**Doctoral Dissertation**

# Controllable neural conversation model considering conversation structure and context

Seiya Kawano

September 21, 2021

Graduate School of Science and Technology
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Science and Technology,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Seiya Kawano

Thesis Committee:

| | |
|---|---|
| Professor Satoshi Nakamura | (Supervisor) |
| Professor Taro Watanabe | (Co-supervisor) |
| Associate Professor Koichiro Yoshino | (Co-supervisor, RIKEN) |
| Professor David Traum | (University of Southern California, USA) |

# Controllable neural conversation model considering conversation structure and context*

Seiya Kawano

**Abstract**

Neural conversation models are end-to-end schemes that generate system responses from user utterances. However, it is challenging to control their response generation. A controllable model based on conditional neural conversation models, which control response generation by conditioning the network on specific intentions considering conversation structures and conversation phenomena, is a promising solution to this problem. This dissertation addresses three problems of conditional neural conversation models.

The first study examines the problem of controllability in conditional neural conversation models. This study considers a conditional neural conversation model where model responses can be controlled by specific intentions considering the conversation structure, such as dialogue acts. By having intentions considering the conversation structure, the system can effectively generate consistent responses towards a dialogue goal. However, current conditional neural conversation models do not sufficiently guarantee to generate high-quality responses that represent the given intention. Therefore, this study proposes a conditional neural conversation model with a new label-aware objective function that promotes generating highly discriminative responses based on the given dialogue acts while maintaining natural responses. Experimental results confirmed that the proposed model generated promising responses in terms of controllability and naturalness compared with those generated from strong conventional models.

*Doctoral Dissertation, Graduate School of Science and Technology, Nara Institute of Science and Technology, September 21, 2021.

The second study incorporates entrainment, an attractive human phenomenon, into neural conversation models. Entrainment is a well-known conversational phenomenon where conversation participants mutually synchronize about various aspects, and is thought to be closely related to human-human conversation quality. Relationships between conversation quality and entrainment were analyzed using an automatic entrainment evaluation measure, and entrainment was shown to improve participant satisfaction in human-human and human-machine conversations. Consequently, a conditional neural conversation model is proposed that can control response generation using a given entrainment degree as the intention. Experimental results showed that the proposed entrainable neural conversation model generated comparable or more natural responses than conventional models, and satisfactorily controlled generated response entrainment.

The third study examined how to automatically understand how intentions are expressed and contribute in practical conversation situations. This study focused on a multi-floor dialogue, i.e., a dialogue that spans multiple conversational floors. Expanding the research scope to multi-floor dialogues will contribute to building autonomous dialogue robots capable of solving real-world problems through multi-floor dialogues. The initial proposed baseline model automatically identifies multi-floor dialogues in an object exploration task in a house. The proposed model's performance was experimentally evaluated, and its limitations and future directions are discussed.

**Keywords:**

Neural conversation model, conditional response generation, dialogue act, entrainment, multi-floor dialogue, dialogue structure parsing

# 会話構造と文脈を意識した<br>制御可能なニューラル会話モデルに関する研究*

河野 誠也


## 内容梗概

ニューラル会話モデルは、ユーザーの発話からシステムの応答を生成するエンドツーエンドモデルとして知られている。しかし、その応答生成を制御することは困難を伴う。この問題を解決するためには、会話構造や会話現象を考慮した特定の意図を条件として応答生成を制御する条件付きニューラル会話モデルの枠組みが有望である。この論文では、条件付きニューラル会話モデルの3つの問題に着目した研究を行った。

一つ目の研究では、条件付きニューラル会話モデルにおける可制御性の問題に着目した。本研究では、会話構造を考慮した特定の意図によって、ニューラル会話モデルの応答を制御する条件付きニューラル会話モデルに焦点を当てる。会話構造を考慮した意図を持つことで、条件付きニューラル会話モデルは会話のゴールに向けて一貫した応答を効果的に生成することが期待できる。しかし、既存の条件付きニューラル会話モデルでは、与えた意図を適切に表現するような応答を生成することを十分に保証しない。本研究では、会話構造を考慮した意図として対話行為に着目し、応答の意図の識別性を考慮した目的関数を備えた条件付きニューラル会話モデルを提案した。提案手法は、生成された応答の自然さを維持しつつ、対話行為の識別性が高い応答の生成を促進する。評価実験の結果、提案モデルは、従来の強力なモデルと比較して制御性と自然性の点で有望な応答を生成できることを示した。

二つ目の研究では、人間が行う魅力的な現象であるエントレインメントをニューラル会話モデルに取り入れる方法について検討した。エントレインメントは、会話における話者間のふるまいが互いに同期する現象であり、人対人における会話の質と密接に関連することが示唆されている。本研究では、まず、エントレインメントを評価するための自動評価指標を用いて、雑談会話における会話の質とエ


*奈良先端科学技術大学院大学 先端科学技術研究科 博士論文, 2021 年 9 月 21 日.

ントレインメントとの関連を分析した。分析の結果、エントレインメントは、人対人、人対機械の会話における、会話参加者の満足度を向上させる可能性があるということを示した。次に、エントレインメント評価指標を用いた制御要素によって、応答を制御可能な条件付きニューラル会話モデルを提案した。提案モデルは、生成文におけるエントレインメントの程度を適切に制御するために強化学習による最適化を適用した。評価実験の結果、提案モデルは、通常のニューラル会話モデルと比較して、応答のエントレインメント度合いを適切に制御できているだけでなく、エントレインメントを考慮することでユーザのシステムに対する満足度を向上できることを示した。

三つ目の研究では、ニューラル会話モデルを含む対話システムが取り扱う対話の問題を拡張した。従来の対話システムは、対話が単一のフロアを持つ場合を想定しており、対話が複数のフロアを持つような場合を想定していない。しかしながら、ある対話で発生した意図を別の対話に伝達し、複数の人間が協働して、共通の目標に従って問題を解決するような状況は我々にとって非常に一般的である。複数フロアの対話に研究の視野を広げることは、複数の人間と会話を通じて、実世界の問題を解決するような自律型ロボットを構築することに寄与すると考えられる。本研究では、最初の調査として、家屋内における物体調査タスクにおけるマルチフロアの対話を対象に、複数フロアの対話がどのような構造を持つのかを自動解析するベースラインモデルの提案を行った。評価実験では、提案モデルの会話構造の解析性能をフロア構造に着目したいくつかの観点から評価し、その限界と研究の将来の方向性について議論した。

**キーワード**

ニューラル会話モデル, 条件付き応答生成, 対話行為, エントレインメント, 複数フロアの対話, 対話構造解析

# Contents

# List of Figures

# List of Tables

# 1 Introduction

## 1.1 Background

Language is the mark of both humanity and sentience, and conversation, or dialogue, is the most fundamental and specially privileged arena of language [Jurafsky and Martin, 2008]. In the Oxford dictionary, a conversation is defined as "A talk, especially an informal one, between two or more people, in which news and ideas are exchanged" Dialogue is defined as "a conversation between two or more people as a feature of a book, play, or film" or "a discussion between two or more people or groups, especially one directed towards exploration of a particular subject or resolution of a problem". However, dialogue and conversation are often used interchangeably. People converse for various purposes, including information exchange, decision-making, and building and maintaining social relationships. A conversation is not just a method for communicating within small groups; the intentions of individual conversations are intertwined in a complicated fashion and can be organized into larger social groups, such as companies, armies, and nations. In other words, society strongly depends on many activities based on conversation. If machines become able to converse with people, they will contribute to activities based on conversation and thereby enrich societies.

In recent years, advances in speech processing and natural language processing technology have led to the rapid development of the *dialogue system*, also called the *conversational agent*, which is intended to converse with humans [Weizenbaum, 1966, Bobrow et al., 1977, Jokinen and McTear, 2009]. Dialogue systems are composed of user interfaces that accept user utterances as input and generate responses as output. Many types of dialogue systems have been developed over the past 50 years, some of which are used worldwide. For example, familiar personal assistants such as Siri* and Cortana† are pre-installed on many smartphones and laptops. An increasing number of dialogue systems are available from other services, such as help desks, airplane ticket booking, and travel guides [Adam et al., 2020]. Dialogue systems have the potential to replace some services provided by human workers.

The common direction of dialogue system research is to imitate human conversation. Conversation between humans is an intricate and complex joint activity. Thus, when we attempt to build a dialogue system that converses with humans, it is important to understand the attributes of human conversation. The first attributes to consider are the principles that humans use in conversation. As one example, this question can be answered by referring to the cooperative principle, which describes how people achieve effective conversational communication in common social situations [Grice, 1975, Davies, 2007]; that is, the cooperative principle describes how listeners and speakers act cooperatively and mutually accept one another to be understood in a particular way. The cooperative principle is divided into the following four conversational elements, as conceptualized in *Grice's maxims*:

1. **Maxim of quality**: Do not say what you believe is false. Do not say that for which you lack adequate evidence.

2. **Maxim of quantity**: Make your contribution as informative as is required (for the current purposes of the exchange). Do not make your contribution more informative than is required.

3. **Maxim of relation**: Be relevant.

4. **Maxim of manner**: Avoid obscurity of expression. Avoid ambiguity. Be brief (avoid unnecessary prolixity). Be orderly.

These maxims are designed to facilitate basic conversations in common social situations. In a conversation, humans can follow Grice's maxims and communicate their intentions naturally, sometimes unconsciously, without being bound by the literal meaning of words and thereby contribute to the conversation as much

*https://www.apple.com/siri/
†https://www.microsoft.com/en-us/cortana

as required. Building a dialogue system that does not violate the principles of conversation that many humans follow, such as Grice's maxims, provides a guide for building a more human-like dialogue system.

The second attribute to consider is how an utterance's intention is being expressed and how it contributes to a conversation. It is important to design a scheme that elucidates how a conversation is structured and how it unfolds, for realizing a human-like conversation in dialogue systems. In computational linguistics, several studies attempted to represent conversations computationally from the viewpoints of both semantic and conversation structure. Semantic structure focuses on the relationship between the meaning of words in an utterance (sentence), such as the predicate-argument structure [Fillmore, 1968]. Over the meaning of words within an utterance, conversation structure focuses on the intentions represented by the utterance and the relationship between series of intentions, such as dialogue acts [Searle, 1965, Austin, 1975, Jurafsky, 1997] and discourse structure [Mann and Thompson, 1988, Prasad and Bunt, 2015]. In particular, conversation structures have traditionally been used to represent an intention of action in dialogue systems. There were also attempts to measure human conversation phenomena such as emotional expression and entrainment, which are consciously or unconsciously included in a conversation, and incorporate them as intentions in dialogue systems [Ball and Breese, 2000, Levitan, 2013, Asai et al., 2020, Levitan, 2020]. By building a dialogue system based on intention and that is conscious of conversation structures and conversational phenomena, we can observe how intentions are expressed and flow through conversations.

The third attribute to consider is how a dialogue system should behave after observing user utterances associated with its conversation history. This attribute relates to a central issue in dialogue systems research. Modular-based dialogue systems, an evolution from frame-based and rule-based systems, are often used by researchers [Oh and Rudnicky, 2000, Jurafsky and Martin, 2008, Ultes et al., 2017]. This framework has both a system intention and an internal state that abstracts information about the conversation history with the user. The system updates that internal state based on the user's utterances, and based on the current internal state, the system chooses an intention for the next action and generates a response. Such pipeline architectures typically include three modules: language understanding, dialogue management, and language generation; each module is built independently using statistical methods and rules. However, building these models independently and on the basis of internal states requires considerable effort.

In recent years, an end-to-end approach based on deep learning has been widely used to construct various types of dialogue systems due to its powerful learning performance [Vinyals and Le, 2015, Serban et al., 2016]. This approach is known as the neural conversation model (NCM). NCMs are based on a sequence-to-sequence model, which learns the mapping between the dialogue history and the system response. Unlike traditional dialogue systems, NCMs are built using a single neural network. Thus, they can reduce the cost of feature engineering and system design compared to traditional systems. Several studies have reported that NCMs can generate responses comparable to humans in chat-style dialogues related to a limited situation [Zhang et al., 2020, Roller et al., 2020]. However, if the system deals with a wide variety of topics or the system have unclear dialogue goals, the system will have difficulty generating consistent and appropriate responses [Gunasekara et al., 2020].

There are also hybrid approaches that combine NCMs with modular-based approaches, in which the response from the NCM is controlled based on the intention provide by the dialogue model (dialogue manager or human heuristics) and dialogue context [Balakrishnan et al., 2019, Dušek et al., 2018]. Unlike the complete end-to-end NCM, such an approach divides the model into the generation and dialogue models, which determines the dialogue strategy or response content. In other words, it borrows the idea of module-based approaches that controlling response generation by specific intentions. In this dissertation, we call such a hybrid generation approach, which has a response control function, as controllable NCMs. Among them, those in which the generation network is extended to accept the dialogue context and specific intentions as their inputs, we call conditional NCMs. By introducing such an approach, developers can focus on creating a qualified dialogue model independent of building a generation model. Furthermore, modeling a high-level decision-making model (such as a dialogue manager or human heuristics) independent of response generation may enable useful generalization and personalization for users. Such a hybrid approach assures the quality of a neural conversation model. In other words, neural conversation models with black-box parameters can be controlled by human developers. In addition, unlike normal NCMs that are entirely end-to-end driven, the system has the potential to reproduce richer human conversations by leveraging intentions that are carefully designed based on human empirical knowledge to control response generation. However, there is still room for improvement in conventional approaches when controlling response generation in conditional NCMs based on specific intentions provided by the dialogue model [Dušek et al., 2018].

This dissertation focus on building controllable NCMs using external conditions. We address three problems associated with constructing conditional NCMs that are driven by specific intentions considering conversation structures and phenomena: 1) the controllability of response generation, 2) the entrainment of response generation, and 3) the understanding of how intentions are expressed and contribute to conversations in practical situations, such as a multi-floor dialogue. In the following sections, we first describe fundamental approaches to building dialogue systems as well as their characteristics. Next, we discuss the three problems faced by current conditional NCMs. Finally, we summarize the approaches and contributions of this dissertation to these problems.

## 1.2 Dialogue system architectures

We can divide dialogue systems into two categories depending on whether the dialogues they handle assume an explicit goal. In a task-oriented system, the system assumes explicit goals, such as a flight search or a tourism guide. It mediates access to databases in natural languages and proactively presents useful information while incrementally recognizing the user's intention through the dialogue. In a non-task-oriented system, the system does not assume explicit goals. The system's only role is to converse with users for enjoyment. There are also open-domain dialogue systems that do not assume specific domains and tasks. Next, we discuss how to construct the dialogue systems described above.

### 1.2.1 Rule-based approach

The use of rules is a simple way to build dialogue systems [Weizenbaum, 1966, Colby et al., 1972]. In general, user utterances are first evaluated based on pre-defined rules such as keyword dictionaries and if-else conditions, among other methods. In rule-based systems, the rule with the highest score is selected to output a pre-prepared response associated with that rule, or a response is generated using a response template. One of the most famous rule-based dialogue systems is ELIZA [Weizenbaum, 1966], which first retrieves keywords that appear in the dialogue history from a hand-crafted dictionary. If a keyword matches, rules are applied to manipulate and transform the user's original utterance, which is then displayed to the user. A few years after the development of ELIZA, another dialogue system focused on clinical psychology; PARRY was developed to study schizophrenia [Colby et al., 1972]. In addition to ELIZA-like rules, the PARRY system included a model of its own mental state, with affect variables for the agent's levels of fear and anger; certain topics of conversation might lead PARRY to become more angry or mistrustful [Jurafsky and Martin, 2008].

ELIZA-style systems [Weizenbaum, 1966, Colby et al., 1972] have been recognized as an important milestone in the development of modern dialogue systems. However, such simple rule-based systems rely on a pre-defined set of rules, and as the system becomes more sophisticated, the number of these rules increases rapidly. Furthermore, simple rule-based systems cannot generate meaningful responses and are only capable of very superficial conversations.

### 1.2.2 Frame-based approach

A frame-based system is a classical approach guided by frames that represent different levels of information in the conversation [Bobrow et al., 1977]. The frames in conversation define the topics of conversation, and the relationships between frames define the flow of conversation. For example, in a conversation about booking an airline ticket, the frames include essential aspects "person," "travel date," "destination," and "time of flight". The simplest frame-based systems are finite state machines, which ask the user a series of pre-defined questions based on the frames. If the user provides an answer, the system moves on to the next question; otherwise, the system ignores that user. Such systems rely on pre-defined frames, ask the user to fill a slot in the frame, and complete the task when all slots in the frame are filled. The limitation of frame-based systems is that generating a response is entirely guided by the information provided by the slots. There is no ability to determine the progress or state of the conversation; for example, the system cannot identify whether the user has rejected a suggestion, asked a question or whether the system now needs to give suggestions or ask questions. In other words, it cannot account for the progress made so far and take appropriate action.

### 1.2.3 Modular-based approach

Modular-based dialogue systems that are more sophisticated versions of simple rule-based and frame-based systems are emerging [Oh and Rudnicky, 2002, Jurafsky and Martin, 2008, Ultes et al., 2017]. Its architecture includes three modules: natural language understanding (NLU), dialogue management (DM), and natural language generation (NLG) (Fig. 1.1). When the system assumes speech-form input and output, automatic speech recognition and speech synthesis modules are applied to pre-processing and post-processing.

The NLU, DM, and NLG modules maintain two key concepts: the "dialogue state," which indicates the current progress of the conversation based on the frame, and "intention," which identifies the characteristics or desired outcome of system action. The NLU module analyzes user utterances and extracts information, including dialogue states that the other modules can understand. The DM module determines the intention of the next system action by considering the information provided by the NLU module. The NLG module generates a system response based on the system intention provided by the DM module using rules, templates, or other statistical methods [Oh and Rudnicky, 2000, Xu and Rudnicky, 2000]. Each module utilizes the dialogue history stored as an internal state when necessary.

The response generation process in a modular-based dialogue system maintains interpretable intentions for characterizing the system action, not only dialogue states (frames). For example, the system explicitly considers speech acts (dia-

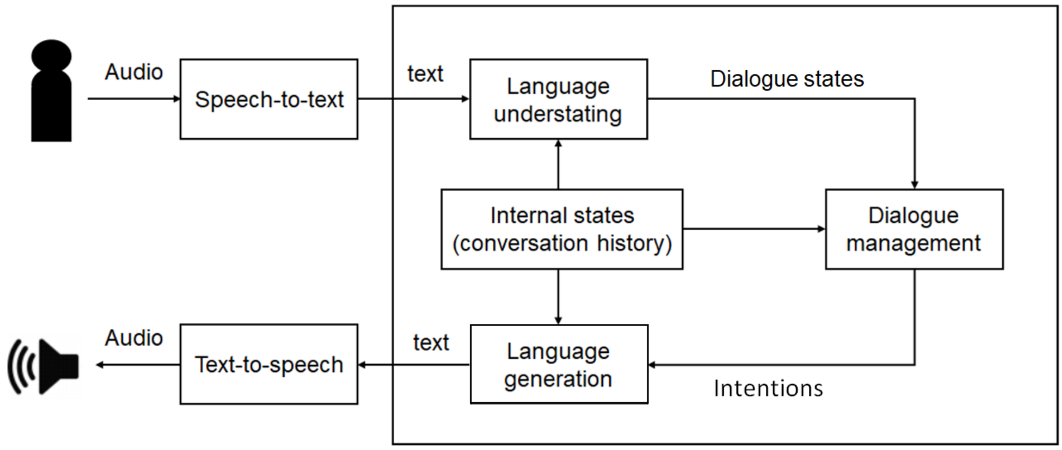



Figure 1.1: Overview of modular-based dialogue system

logue acts), discourse structures (discourse relations), emotions, dialogue topics, and contents words. Alternatively, the system considers a set of slot-value that integrates such various attributes as a frame. They are mainly used in DM and function as control factors in modular-based dialogue systems. Some studies modeled DM, which is the core module of modular-based dialogue systems, by incorporating such intentions as output and dialogue states (or other representation of dialogue context) as input, based on statistical methods, such as a partially observable Markov decision process (POMDP) [Williams et al., 2008, Young et al., 2010, Meguro et al., 2010, Yoshino and Kawahara, 2015]. When building a dialogue system, especially a DM, dialogues must be thoroughly analyzed and characterized. Considering such information as the participant's intentions and the purposes behind their utterances, its relationship with them is critical to determine coherent system actions for addressing goals or sub-goals in dialogues. This problem is also related to the core issues addressed in such fields as pragmatics, and we can take advantage of the heuristics known in those fields [Verschueren, 2005]. Speech acts and the discourse structure that are considered conversation structures described in the next section are useful when addressing this problem.

The advantages of a modular-based system are that the system's goal is explicitly defined and that the pre-designed dialogue state and intention provide a clear guideline for how a conversation should proceed. On the other hand, its limitation is that building these modules independently depending on the design of internal states requires considerable effort.

### Speech acts in dialogue

In the philosophy of language and pragmatics, a speech act is known as a type of act that can be performed by a speaker meaning that one is doing so [Austin, 1975, Searle, 1965]. Another name is a dialogue act, which is a suitable concept for describing any dialogue system communication, not just human-human communication [Bunt, 2006, Traum, 1999]. In speech act theory, a speech act is divided into the following three levels:

- A locutionary act: the performance of an utterance and hence of a speech act.
- An illocutionary act: the active result of an implied request or a meaning presented by the locutionary act. For example, if the locutionary act in the interaction is the question *Is there any salt?*, the implied illocutionary request is *Can someone pass the salt to me?*.
- A perlocutionary act: the actual effect of such locutionary and illocutionary acts as persuading, convincing, scaring, enlightening, inspiring, or otherwise getting someone to do or realize something, intended or not.

In particular, an illocutionary act is an analytical level widely used to analyze dialogues and develop computational models of dialogue. The analytical level of illocutionary acts can be easily understood by humans, and their annotation cost is low. In modular-based dialogue systems, an illocutionary act is known as a dialog act, and it is the key unit of each module. In particular, DM characterizes the behavior of dialogue systems at the dialogue act level. It is a central challenge in modular-based dialogue systems. To address this challenge, much research has examined how to determine dialogue acts because they can contribute to the dialogue's goal in a given dialogue context, in both task-oriented and non-task-oriented systems [Williams et al., 2008, Young et al., 2010, Meguro et al., 2010, Yoshino and Kawahara, 2015]. In other words, these studies optimized a response strategy of dialogue systems at the level of dialogue acts.

### Discourse structure in dialogues

In general, DMs determine system actions by considering the structure of dialogues using a concept that abstracts over utterances such as speech acts. Since a speech act is a concept that focuses on the actions performed by utterances, it is not easy to explicitly track how each utterance in a dialogue contributes to its goal. We can also employ the level of discourse structure to consider the dialogue's structure. Discourse relations, also called *coherence relations* or *rhetorical relations*, are relations (expressed explicitly or implicitly) between situations mentioned in discourse and are key to a complete understanding of the discourse (dialogue) beyond the meaning conveyed by sentences (utterances) [Bunt and Prasad, 2016].

Rhetorical structure theory (RST) is one popular discourse structure theory, which addresses what relations hold between utterances and how they are linked to form a connected whole in terms of the speaker's intentions and effects on the listener [Mann and Thompson, 1988]. RST is defined by a ternary relationship among the main element (nucleus), a supporting element (satellite), and their rhetorical relations. RST is a fundamental concept for describing discourse structures, and several derivatives have been proposed [Taboada and Mann, 2006]. Discourse structure parsing identifies such ternary relationships, and its parsing results are utilized in DM. By taking into account the discourse structure, the DM can logically understand the transitions of the states of dialogues and decide coherent actions. Discourse structure theories are also often applied to the development of various NLP applications, such as summarization, question answering, and argumentation mining [Uzêda et al., 2010, Verberne et al., 2007].

## 1.2.4 Non-modular-based approach

We can categorize non-modular-based approaches into retrieval- and generation-based systems, such as neural conversation models.

### Retrieval-based system

A retrieval-based system integrates NLU, DM, and NLG modules into one core module that functions as a response selection module [Lee et al., 2009, Kim et al., 2010]. In general retrieval-based systems, the system response is chosen from pairs $\mathbf{e}$ of query utterance $q$ and response utterance $r$, constructed from actual dialogues. When query utterance $q$ is the most similar to the system input (user utterance) $q'$ the response utterance $r$ that corresponds to $q$ is selected as system response $r'$:

$$\langle q', r' \rangle = \arg \max_{\langle q,r \rangle \in \mathbf{e}} \mathrm{sim}(q', q). \tag{1.1}$$

The advantages of retrieval-based systems are that the responses are grammatically correct and lexically diverse since they are just copied from the set of context-response pairs. The disadvantages of retrieval-based systems are that the system's performance depends on the size of the set of context-response pairs, and if the set does not contain any pairs with contexts similar to the observed context, the system will not work correctly [Gandhe and Traum, 2010, Song et al., 2018]. Furthermore, these systems need carefully designed similarity functions and re-ranking functions to identify subtle semantic differences between different input contexts.

**Generation-based system**

In contrast with retrieval-based systems, generation-based systems create responses word-by-word instead of copying responses from a training set. This task can be formalized as an input-output mapping problem for which, given a dialogue history of utterances, the system must output a coherent and meaningful sequence of words.

To address this problem, early work applied phrase-based machine translation to response generation [Ritter et al., 2011]. However, phrase-based translation complicates translating user input and system output since it has many different components. This problem also applies to machine translation tasks, but it is even more serious for response generation tasks. Unlike machine translation, in which there is usually an obvious mapping between the source and target sentences, the mapping between user inputs and system outputs is less obvious. Therefore, phrased-based translation is only adept at handling the few cases in which word-level mapping is obvious; response generation usually generates inconsistent or ungrammatical responses when the input sentences' semantics become complex.

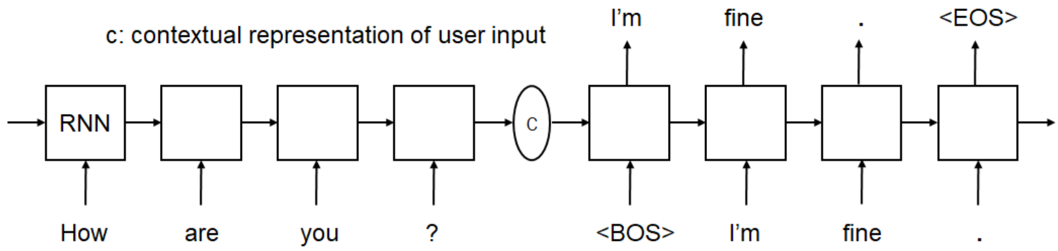



Figure 1.2: Architecture of encoder-decoder model

Recent progress in machine translation emerged from neural language models [Mikolov et al., 2011] and encoder-decoder models [Sutskever et al., 2014]; this progress led to a neural-based end-to-end response generation model called a *neural conversation model* [Vinyals and Le, 2015, Serban et al., 2016]. The NCM, which is based on the encoder-decoder model, includes both encoder and decoder networks based on the recurrent neural network (RNN) (Fig. 1.2). The encoder transforms a user utterance into a contextual representation, which is a fixed-length vector. The decoder recursively generates the response's word sequence using the contextual representation of the user utterance as an initial hidden state for a decoder RNN. However, since decoders do not consider the dialogue history, they often generate unnatural responses. To address this problem, several extensions have been proposed to handle contextual dialogue information efficiently. One study proposed an extended architecture for encoder networks, which uses a hierarchical encoding strategy that considers both the levels of utterances and contexts [Serban et al., 2016].Other studies proposed both attention and copying mechanisms that consider the efficient connection of encoders and decoders based on the dialogue history [Xing et al., 2018, Eric and Manning, 2017]. Since the language generation architecture for dialogue systems is also studied from many other perspectives, it continues to evolve.

NCMs have the potential to generate flexible and tailored responses, which distinguishes them from retrieval-based systems that just copy from a training set [Song et al., 2018]. However, NCMs often generate less diverse responses, in contrast to retrieval-based systems. NCMs and retrieval-based systems both suffer from the size of the training set[‡]. The advantages and disadvantages of NCMs and retrieval-based systems are complementary to each other, and several approaches that combine the two approaches have been proposed s [Song et al., 2018, Yang et al., 2019]. However, both retrieval-based systems and NCMs are key approaches for building dialogue systems. This dissertation focuses on the problems faced by NCMs.

### 1.2.5 Hybrid approach

Dialogue systems can be built by combining both modular- and non-modular-based architectures [Pichl et al., 2018, Finch et al., 2020, Paranjape et al., 2020]. This combination is often seen, for example, in systems that compete in the Alexa Prize Challenge[§]. In this challenge, each team builds a social dialogue system that talks to volunteers on crowdsourcing services; these systems are then evaluated based on the conversation length and user ratings. For example, Chirpy Cardinal[¶] uses information fed from NLP components such as Wikipedia entity links, user intent classification, and dialogue act classification to determine the intention of system action by a high-level decision-making model. The system's response is generated from one of a series of response generators (e.g., rules, retrievals, and neural conversation models, etc.) according to the intentions provided by the high-level decision-making model. Such modular design allows users to add new response generators or change the design of high-level decision-making models without changing large parts of the codebase every time they expand the system's coverage.

In this dissertation, we focus on a variant of NCMs that employs a hybrid approach: conditional neural conversation models (conditional NCMs). Conditional NCMs control their response generation according to the intention provided by a high-level decision-making model, not only the dialogue context. In general, NCMs cannot control the content of generated responses. Conditional NCMs ad-

[‡]When attempting to build a non-task-oriented dialogue system, we can use a large amount of data from SNS, such as Twitter and Reddit.
[§]https://developer.amazon.com/alexaprize
[¶]https://web.stanford.edu/ jurafsky/slp3/

dress this problem by extending the generation network to accept interpretable intentions, not only dialogue contexts as their inputs. For example, dialogue acts, emotions, personas, and topic information can be explicitly input to the model as system response characteristics. This process makes it possible to characterize the system's response generation direction, improve low response diversity in NCMs, and enhance models' interpretability by having human-interpretable intentions. In addition, unlike normal NCMs that are entirely end-to-end driven, the system has the potential to reproduce richer human conversations by leveraging intentions that are carefully designed based on human empirical knowledge to control response generation. Another approach is a conditional neural language model, which is conditioned by a frame (a set of slots and values) that characterize the system response characteristics for response generation [Wen et al., 2015b, Wen et al., 2015a]. It reduces the cost of constructing the pre-defined templates and rules, which is a weakness of traditional template-based language generation. However, its training framework requires frames that entirely express the function and the contents of the target response utterances. The boundary between the two approaches is unclear, but conditional neural language models based on the frame focus on task-oriented dialogues. In contrast, conditional NCMs focus on both task- and non-task-oriented dialogues.

There are NCMs that use not only dialogue contexts but also additional external inputs (such as user's emotion and background knowledge), although they do not explicitly control the response by the system's intention. To avoid confusion, note that "conditional" in this dissertation does not simply mean that the additional input is used but that the additional input represents the system's intention of response. We can also use controllable but not conditional NCM. For example, we can select the response that best realizes the specified intention from the candidate responses of an NCM trained with only context-response pairs. Such an approach can also be applied to conditional NCMs. However, this is beyond the scope of this dissertation.

## 1.3 Problems

In this dissertation, we focus on building a conditional NCM driven by specific intentions. As mentioned earlier, intentions that account for conversation structure, such as dialogue acts and discourse structure, which idea of modular-based dialogue systems, can help the system generate consistent responses. However, due to the complex structure of natural language, it is still a challenge for a system, given such intentions, to express them adequately and produce responses that are natural to the context. This problem also makes it more difficult to leverage the intentions provided by the high-level decision-making model or the human heuristics. It is also difficult to explicitly build a different intention that considers conversational phenomena that are attractive to humans, such as entrainment, into the system. This complexity arises because, in addition to the problem of response generation controllability, the system is limited in its ability to exploit context. Furthermore, automatic understanding of how intentions are expressed and contribute to conversations in practical situations, such as in multi-floor dialogue, is important for developing high-level decision-making models and exploring dialogue systems' future direction, including neural conversation models.

In this section, we describe three problems in conditional neural conversation models: 1) the controllability of response generation, 2) the entrainment of response generation, and 3) multi-floor dialogue.

### 1.3.1 Controllability of response generation

Early work with conditional NCMs incorporated a persona into the system to ensure the NCM's response consistency [Li et al., 2016b]. Researchers prevented the system from generating responses that contradicted its personas, including attributes such as hobby and age. Furthermore, there are several studies on controlling NCM response generation based on emotions [Huang et al., 2018a, Zhou et al., 2018]. Their systems create a positive impression on the user by generating responses based on various emotional expressions. Several studies attempted to control NCM responses based on the conversation structure, such as dialogue acts [Wen et al., 2015b, Wen et al., 2015a, Dušek et al., 2018]. These approaches aim to enhance the quality of NCMs in terms of discourse coherence and cohesion as well as make NCMs easier to control. Conditional language generation based on various intentions is a central issue in recent data-driven NLG research.

Various sophisticated network architectures were proposed for conditional NCMs. Several studies enhanced their performance by introducing attention and gating mechanisms, which usually use training objectives to minimize the cross-entropy loss of word prediction during the response generation process corresponding to the given dialogue history [Xing et al., 2017, Zhou et al., 2018]. However, such training objectives often lead to responses that are biased by frequent words or phrases in the training data: *I don't know, Me too,* and *Yes please.* These frequent responses are examples of the dull response problem [Li et al., 2016a]. Furthermore, the response does not necessarily represent the semantics of the given condition, even if it contains words or phrases that represent the characteristics of a given condition and are relevant to the context. This problem is also related to conversational implicature (for dialogue acts) and irony (for emotions). However, this semantic complication is often ignored in existing studies, which judge results using automatic evaluation based on relevance with references such as BLEU.

Thus, we must explore a new training objective function for conditional NCMs that improves both the naturalness and the controllability of response generation.

### 1.3.2 Entrainment of response generation

Entrainment is a conversational phenomenon in which dialogue participants mutually synchronize various aspects: lexical choice [Brennan and Clark, 1996], syntax [Reitter and Moore, 2007], style [Niederhoffer and Pennebaker, 2002], acoustic prosody [Natale, 1975, Ward and Litman, 2007], turn-taking [Campbell and Scherer, 2010, Beňuš et al., 2014], and dialogue acts [Mizukami et al., 2016]. Entrainment has been studied in various fields and is also known as convergence, coordination, alignment, or synchrony.

In social psychology, entrainment is related to "convergence" and "divergence" in speech accommodation theory (SAT), which describes the mechanism of language use in terms of whether the speaker wants to approach or distance himself from a speech partner [Giles et al., 1987, Giles, 2016]. Convergence refers to the process through which an individual shifts their speech patterns in an interaction to more closely resemble their conversation partner's speech patterns. One individual shifts him/her speech to assimilate to the other; this action can result in a more favorable appraisal when convergence is perceived positively [Giles, 1979]. In contrast, divergence is a linguistic strategy whereby a speaker accentuates the linguistic differences between himself and his interlocutor [Giles et al., 1991]. Given that communication features often compose the identity of a group member, divergence is a crucial tactic for displaying distinctiveness from others. Divergence is often used to maintain one's own identity as part of a different group or maintain autonomy from a group. A speaker's speech pattern depends on who they are talking to and how they feel about that person. For example, a situation in which person A says "toilet" and person B says "restroom" could arise for various reasons: B may have emphasized the difference in dialects to A (British English and American English) to maintain his identity as a member of a different group. In the case that A and B belong to the same group, B may have used a different word to display his authority or identity within that group. It may simply be that B is familiar with the word "restroom" and used it unconsciously. The motivations for using convergence and divergence are diverse, and they depend on the speaker's attributes and relationship to the other speaker.

SAT qualitatively argued that entrainment (convergence) is closely related to building a rapport with others in conversation. Several studies also analyzed entrainment from a quantitative perspective and reported that it has a high correlation with dialogue success, naturalness, and engagement in conversation [Nenkova et al., 2008, Levitan et al., 2015, Nasir et al., 2019]. These studies suggest that entrainment has the potential to make dialogue systems more natural and attractive. However, to the best of our knowledge, no previous studies have explicitly incorporated such phenomena into NCMs.

Two problems surface when incorporating entrainment into NCMs. The first problem is that NCMs suffer from dull response issues. In other words, dialogue systems suffer from using context before generating entrained responses. The second problem concerns how much of an entrained response should be generated for the context. The simplest strategy is to trace the user's conversational style and lexical choices completely. However, excessive entrainment leads to negative user impressions. Furthermore, we also need to consider convergence, which moves away from the user's speech pattern, may be necessary in some situations. Thus, we require a framework that can arbitrarily control the degree to which responses are entrained, as the intention of system response. By introducing such a framework, we can use empirical rules to dictate the degree of entrainment or use the output from a high-level decision-making model that theoretically determines the degree of entrainment.

### 1.3.3 Multi-floor dialogue

The floor is defined as the acknowledged what's-going-on within a psychological time/space [Edelsky, 1981]. What's going on can be the development of a topic or a function (teasing, soliciting a response, etc.) or an interaction of the two. It can be developed or controlled by one participant at a time or by several simultaneously or in quick succession. A floor is also closely related to the concept of turn-taking, which recognizes who is taking the turn (the right to speak) in conversation [Sacks, 1992]. Although holding turn and holding floor are sometimes used in similar meaning, the scope of the definition of the floor is usually discussed as something beyond the turn [Edelsky, 1981]. For example, dialogue participants might describe a floor thus: "Jack's telling us about his holidays" (in which case Jack holds the floor); or "We're chatting about the film we've just watched" (in which case the floor is jointly produced) [Herring et al., 2013]. In other words, holding the turn does not necessarily equate to holding the floor. A dialogue with two or more participants usually requires a single floor to advance.

In contrast, when two or more floors exist in parallel, multiple dialogue floors are evident [Cherny et al., 1999]. For example, although an internet relay chat (IRC) has a single message stream, multiple participants might simultaneously chat about different topics. An individual participant may be involved in more than one dialogue floor in such a case because the dialogue content is visible to all participants. However, these situations are particular to modern society, where the internet has become pervasive.

In this dissertation, we are interested in a case called *multi-floor dialogue*, in which a specific intention is realized across multiple floors, and only a subset of participants can join different floors [Traum et al., 2018]. A multi-floor dialogue consists of multiple sets of dialogue participants, each conversing within their own floor; however, at least one multi-communicating member who is a participant in multiple floors coordinates with each to achieve a shared dialogue goal. For example, in a restaurant, a server communicates with customers and takes their orders in the dining room (one floor); the server also talks with other workers in the kitchen (another floor). All of the participants work toward the joint goal of providing the customer with their desired meals. However, in this case, only the server participates on both floors, conveying orders from customer to kitchen and perhaps information about item availability or speed from the kitchen back to the customers. Another example is found in military units; soldiers follow their commander's orders, which are decided at headquarters. Such situations are quite common in the real world, where there are dialogue floors for decision-making and other floors that implement actions based on those decisions.

Until now, various dialogue systems, including neural conversation models, did not assume that a dialogue has multiple floors. Although an annotation scheme that describes the structure of multi-floor dialogues was proposed [Traum et al., 2018], building a computational model that addresses multi-floor dialogues remains limited. Implementation obstacles stem from the continuing unavailability of a language understanding model that automatically identifies the structure of multi-floor dialogues. By building a model that can automatically identify this structure, we promote the development of language resources and computational models for cooperative dialogue systems. Such development would also improve communication protocols through natural language by analyzing how dialogue intentions are communicated on different floors.

## 1.4 Approaches and contributions

In this dissertation, we address three problems associated with constructing conditional NCMs that are driven by specific intentions considering conversation structures and phenomena:

- In the first study (Chapter 3), we examined how to improve the controllability of response generation in a conditional NCM that is constrained by intentions considering conversation structure. We aimed not only to reflect the given intentions in the generated responses but also to improve the naturalness of the generated responses when compared to conventional methods.

- In the second study (Chapter 4), we explored how to incorporate attractive attributes of human-human conversations into conditional NCMs. We focused on entrainment, a well-known conversational phenomenon in which dialogue participants mutually synchronize various conversational components. We then build a conditional NCM that uses the degree of entrainment as the system's intention for response generation. We aimed to intentionally create these attractive phenomena in dialogues so that the system induces higher user satisfaction.

- In the third study (Chapter 5), we examined automatic understanding of how intentions are expressed and contributed to practical conversation situations. We focused on multi-floor dialogue. We aimed to contribute to a cooperative dialogue system by building a dialogue structure parser that automatically identifies the structure of multi-floor dialogues.

Our studies borrow the idea of module-based dialogue systems that control response generation by specific intentions, carefully designed by human empirical knowledge. In particular, we focused on types of intentions considering conversation structures and phenomena. By considering such intentions in NCMs, they probably can realize more engaging conversations with users. An overview of this dissertation is shown in Fig. 1.3[‖]. Our studies are related to the NLU and NLG methods in dialogue system architecture. Building a high-level policy decision model (such as DM module) for deciding system intentions and integrating each study is a future challenge. The approaches and contributions of each study are described in the following sections.

[‖]Note that since our approach borrows ideas from modular-based systems, we have mapped our contributions to each module of a modular-based system for a brief description.

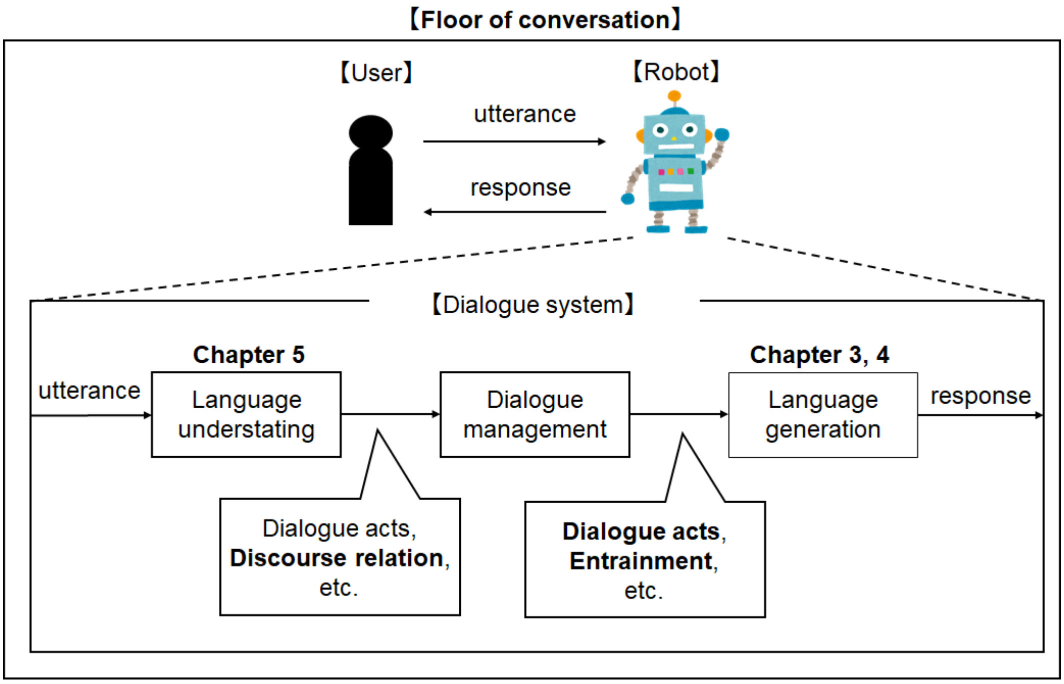



Figure 1.3: Dissertation overview

### 1.4.1 Controllable neural conversation model by given dialogue acts

This study addressed the problem described in Section 1.3.1. We focused on controlling the responses of NCMs using intentions that consider dialogue structure, especially the possibility of using dialogue acts as intentions. Dialogue acts are often used as an interpretable unit that analyzes and models both human-human and human-machine dialogues. If we can control NCMs using dialogue acts, the systems can generate consistent responses by considering conversation structure.

We introduced a reinforcement learning framework for conditional response generation that involves adversarial learning. Our proposed method has a new label-aware objective that encourages discriminative responses using the given dialogue act labels while maintaining the naturalness of the generated responses. We compared the proposed method with conventional methods that generate conditional responses, and the experimental results demonstrated that our proposed method has higher controllability even though its naturalness is better than or comparable to conventional models. Our proposed model has the potential to be applied to other intentions besides dialogue acts.

### 1.4.2 Entrainable neural conversation model

This study addressed the problem described in Section 1.3.2. We focused on entrainment, a well-known conversational phenomenon in which conversation participants mutually synchronize various conversational components. Entrainment has a high correlation with dialogue success, naturalness, and engagement. In general, humans consciously and unconsciously create entrainment in conversation to improve that quality. Incorporating entrainment into neural conversation models has the potential to create a more human-like system.

In this study, we defined entrainment scores based on word similarities in semantic spaces to evaluate the system's entrainment. We optimized an NCM's entrainment scores using reinforcement learning to control the system response's degree of entrainment. In other words, the system's intention is based on the degree of the response's entrainment to context. The experimental results demonstrated that the proposed entrainable neural conversation model generated comparable or more natural responses than conventional models and satisfactorily controlled the degree of entrainment of the generated responses.

### 1.4.3 Dialogue structure parsing on multi-floor dialogues

This study addressed the problem described in Section 1.3.3. Automatically understanding how intentions are expressed and contribute to conversations in practical situations such as in multi-floor dialogue is important for developing high-level decision-making models and for the future direction of dialogue systems, including neural conversation models. Expanding the research scope to multi-floor dialogues will contribute to building cooperative dialogue robots that solve real-world problems by modeling dialogues across multiple floors. We first automatically identified how participants, including robots, would proceed in a dialogue related to such domains as urban search and rescue or military reconnaissance.

In this study, we proposed a baseline model that automatically identifies the multi-floor dialogue structure based on multi-task learning and an attention mechanism. We used a multi-floor dialogue dataset annotated with a discourse structure, which was created as part of a long-term project to develop an autonomous robot by remote human participants. In the experiment, we showed that our proposed model has a promising identification performance for dialogue structure and discussed its limitations and the future direction of this study.

## 1.5 Dissertations organization

The following is the organization of this dissertation. First, we introduce the general architecture of a neural conversation model and its technical backgrounds in Chapter 2. In Chapter 3, we describe a conditional neural conversation model that can control the generated response according to the given dialogue act. In Chapter 4, we describe a conditional neural conversation model that can control the entrainment degree of the generated response based on the given entrainment degree. In Chapter 5, we describe a dialogue structure parser that identifies the structures of multi-floor dialogues. Finally, we conclude and discuss future directions in Chapter 6.

# 2 Sequence-to-Sequence model for response generation

In this chapter, we describe the sequence-to-sequence model, which is a fundamental approach for building neural conversation models.

## 2.1 Neural language model

Recent progress in language generation studies has been based on recurrent neural network language models (RNNLMs) [Mikolov et al., 2011]. RNNLMs differ from the previous n-gram based language models in that they use a state vector as a memory of the past. The state vector is updated at each time-step and thus can capture an unbounded history, in theory. The RNNLM calculates the occurrence probability of a given word sequence $Y = [y_1, y_2, \cdots, y_T]$ as:

$$\begin{aligned} p(y_{1:T}) &= p(y_1, y_2, ..., y_T) \\ &= \prod_{t=1}^{T} p(y_t | y_1, y_2, ..., y_{t-1}) \end{aligned} \tag{2.1}$$

Here, $T$ is the length of $Y$, $y_t \in \mathcal{V}$ is the output word at time-step $t$, and $\mathcal{V}$ is the vocabulary. In the RNNLM, the conditional probability $p(y_t|y_{1:t-1})$ is recursively calculated as:

$$h_t = \mathrm{RNN}(y_{t-1}, h_{t-1}) \tag{2.2}$$

$$h_t^o = \mathrm{Linear}(h_t) \tag{2.3}$$

$$p(y_t|y_{1:t-1}) = \mathrm{Softmax}(y_t, h_t^o) \tag{2.4}$$

Here, RNN is a nonlinear function with trainable parameters, which can be used for various types of recurrent neural networks (RNNs). Linear$(\cdot)$ is a linear transformation function with trainable parameters, which transforms $h_t \in \mathbb{R}^{h_d}$ into a fixed-size vector $h_t^o \in \mathbb{R}^{|\mathcal{V}|}$. Softmax$(\cdot)$ is a function that calculates the probability distribution of $y_t$ corresponding to a given $h_t^o$.

### 2.1.1 Recurrent neural networks

The vanilla recurrent neural network (RNN) model is a neural network architecture designed to handle sequential data, such as sentences (word sequences). The RNN calculates a hidden vector representation $h_t$ associated at each time step $t$, which can be considered a representation that embeds information of previous words. $h_t$ is calculated using a nonlinear function that combines both the previously built representation $h_{t-1}$ and the $K$-dimensional input vector $x_t \in \mathbb{R}^K$ which is associated with $y_{t-1}$ as:

$$h_t = \sigma(W_{hh} \cdot h_{t-1} + W_{xh} \cdot x_t + b_{hh} + b_{xh}) \tag{2.5}$$

Here, $W_{hh} \in \mathbb{R}^{h_d \times h_d}$ and $W_{xh} \in \mathbb{R}^{h_d \times K}$ are weight matrixes, and $b_{hh} \in \mathbb{R}^{h_d}$ and $b_{xh} \in \mathbb{R}^{h_d}$ are biases. $\sigma$ is an activation function, such as sigmoid, tanh, or ReLU.

However, vanilla RNN has a problem called the vanishing gradient problem. Gradients vanish/explode during back-propagation, and as a result learning does not go well. This is because sequential data is often long, and the RNN becomes deep in proportion to the length of the sequential data [Bengio et al., 1994].

### 2.1.2 Long short-term memory networks

Long short-term memory (LSTM) has been proposed to address the vanishing gradient problem of RNNs [Hochreiter and Schmidhuber, 1997]. The difference from a RNN is that LSTM replaces units with memory units, which tune the unit value $c_t$ and the output $h_t$ using gating mechanisms (input gate, memory gate, output gate) over time. The hidden vector $h_t$, which is the output of LSTM, is calculated as:

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i) \tag{2.6}$$

$$f_t = \sigma(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \tag{2.7}$$

$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \tag{2.8}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tanh(W_c \cdot x_t + U_c \cdot h_{t-1} + b_c) \tag{2.9}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{2.10}$$

Here, $i_t \in \mathbb{R}^{h_d}$ is the input gate, $f_i \in \mathbb{R}^{h_d}$ is the forget gate, $o_t \in \mathbb{R}^{h_d}$ is the output gate, and $c_t \in \mathbb{R}^{h_d}$ is the is the memory cell. $W \in \mathbb{R}^{h_d \times K}$ and $U \in \mathbb{R}^{h_d \times h_d}$ are the weight matrixes, and $b \in \mathbb{R}^{h_d}$ are the biases. Instead of LSTM, other architectures can be used to address gradient loss problems, such as the gated recurrent unit (GRU) [Cho et al., 2014].

## 2.2 Encoder-decoder model

The encoder-decoder model, also called the sequence-to-sequence model, is a neural network model that learns the transformation of a source sentence $X = [x_1, x_2, \cdots, x_T]$ into a target sentence $Y = [y_1, y_2, \cdots, y_{T'}]$. It can be regarded as a conditional neural language model conditioned by a dialogue context $X$ for generating a system response $Y$. The encoder-decoder model usually consists of three modules: encoder, decoder, and attention mechanism.

### 2.2.1 Encoder

The encoder encodes the word inputs $X = [x_1, x_2, \cdots, x_T]$ into vector representations $[h^x_1, h^x_2, \cdot, h^x_T]$ using a RNN, as:

$$h^x_t = \mathrm{RNN}(x_t, h_{t-1}) \tag{2.11}$$

Here, the initial hidden state $h^x_0$ is usually given as a zero-vector. Then the last hidden state $h^x_T$ is fed into the decoder network.

Various forms of input can also be encoded by extending the above encoder network, for example, the input formed by dialogue history, including several utterances, based on an encoder that uses a hierarchical encoding strategy, which considers both the levels of utterances and contexts [Serban et al., 2016].

### 2.2.2 Decoder

The decoder computes a target sentence $Y = [y_1, y_2, \cdots, y_{T'}]$ using the last hidden state $h_T^x$ generated by the encoder, as:

$$h_t^y = \mathrm{RNN}(y_{t-1}, h_{t-1}^y) \tag{2.12}$$
$$h_t^o = \mathrm{Linear}(h_t^y) \tag{2.13}$$
$$p(y_t|y_{1:t-1}) = \mathrm{Softmax}(y_t, h_t^o) \tag{2.14}$$

Here, the initial hidden state $h_0^y$ is the last encoder state $h_T^x$, and the first decoder input $y_0$ is a special token that indicates the start of the target sentence. The decoder samples $y_t$ from the probability distribution $p(y_t|y_{1:t-1})$ to use as the input for the next step and continues generating until the model outputs a token that indicates the end of the sentence.

### 2.2.3 Attention-based decoder

The above decoder network can be extended by using an attention mechanism for encouraging the use of information in the source sentence [Luong et al., 2015]. The attention mechanism considers all the hidden states of the encoder when deriving the output vector $h_t^o$ for each time-step $t$, as:

$$\alpha_t(s) = \frac{\exp(\mathrm{score}(h_t^y, h_s^x))}{\sum_{s'} \exp(\mathrm{score}(h_t^y, h_{s'}^x))} \tag{2.15}$$
$$c_t = \sum_s \alpha_t(s) * h_s^x \tag{2.16}$$
$$\tilde{h_t^y} = \tanh(\mathrm{Linear}([c_t; h_t^y])) \tag{2.17}$$
$$h_t^o = \mathrm{Linear}(\tilde{h_t^y}) \tag{2.18}$$

Here, $\mathrm{score}(\cdot)$ is any function that calculates the score of both $h_t^y$ and each hidden vector of the encoder. This step helps the decoder find relevant information on the encoder side corresponding to the decoder's current hidden states.

## 2.3 Training objectives

Sequence-to-sequence models, such as encoder-decoder models, can be optimized in several ways. In general, methods based on maximum likelihood estimation and reinforcement learning are used.

### 2.3.1 Maximum likelihood estimation

Sequence-to-sequence model training is usually performed using a teacher-forcing approach. This approach predicts an output by feeding the oracle output from the previous time-step into the decoder. It does not use the predicted output word as the decoder input in the next time-step. Teacher forcing allows the training to converge. The entire network is trained by minimizing the negative log-likelihood (cross-entropy loss) of predicting each word in the target $Y$ given the source $X$, as:

$$\mathcal{L}(\theta) = -\frac{1}{T'} \sum_{t}^{T'} \log[p(y_t|y_{1:t-1}, X)] \tag{2.19}$$

Here, $p(y_t|y_{1:t-1})$ is the output probability of $y_t$ at time-step $t$, calculated by the softmax function, and $y_t$ is the actual word label.

### 2.3.2 REINFORCE algorithm

For applying reinforcement learning to sequence-to-sequence models, a Markov Decision Process (MDP) defined by the environment and states, the agent's actions, the reward function, and the state's transition probabilities is used.

The MDP's action space is the target response's vocabulary, actions are words, and the states are the decoder's hidden vectors. The state's transition probabilities are implied by the operation from the RNN cell inside the decoder. The decoder starts the generation with an initial state $h_0$, representing the user input $X$ computed by the encoder. At any time step $t$, the decode decides the next action to take by defining a stochastic policy $\pi(y_t|h_{t-1}, X)$, which takes the previously hidden state vector $h_{t-1}$ as input and produces a probability distribution over all actions that are defined by the words in the target sentence vocabulary.

The next action $y_t$ is chosen either by sampling from this policy or by taking the argmax. The model computes the next state $h_t$ by updating the current state $h_{t-1}$ by the action taken $y_t$.

The policy-based approach directly optimizes parameterized $\pi_\theta$ to maximize the expected reward. $\theta$ is a neural network parameter of sequence-to-sequence models. The policy gradient is one of the algorithms that optimize parameterized policy $\pi_\theta$ with respect to the expected reward return, as:

$$J(\theta) = \mathbb{E}_{\hat{y}_1,\cdots,\hat{y}_{T'} \sim \pi_\theta(\hat{y}_1,\cdots,\hat{y}_T)} \left[ r\left(\hat{y}_1, \cdots, \hat{y}_{T'}\right)\right] \tag{2.20}$$

$$\theta \leftarrow \theta + \alpha \nabla_\theta J(\theta) \tag{2.21}$$

Here, $\alpha$ denotes the learning rate for stochastic gradient decent.

The major challenge involved in policy gradient methods is obtaining a good estimate of the policy gradient $\nabla J(\theta)$. Based on the policy gradient theorem [Williams, 1992], REINFORCE Algorithm determines the gradient of an objective function for parameter $\theta$, as:

$$\nabla_\theta J(\theta) = \mathop{\mathbb{E}}_{\hat{Y} \sim \pi_\theta} \left[ \nabla_\theta \log \pi_\theta(\hat{Y}) \times (r(\hat{Y}) - b) \right] \tag{2.22}$$

This method is that the model suffers from high variance, when only one sample is used for training at each time step. To address this challenge, at each training step, one can sample $M$ sequences of actions and update the gradient by averaging over all these $M$ sequences, as:

$$\nabla_\theta J(\theta) = \frac{1}{M} \sum_{i=1}^{M} \sum_{t=1}^{T'} \nabla_\theta \log \pi_\theta(\hat{y}_{i,t} | \hat{y}_{i,t-1}, h_{i,t-1}) \times (r(\hat{Y}_i) - b) \tag{2.23}$$

Here, baseline $b$ can be any arbitrarily chosen scalar, because it does not introduce bias in the gradient. Suggested values of baseline $b$ include the mean value of all previously observed rewards, or estimator output from another neural model [Zaremba and Sutskever, 2015, Ranzato et al., 2015].

# 3 Controllable neural conversation model by given dialogue acts

In this chapter, we describe a conditional neural conversation model that can control the generated response by given intention aware of conversation structure.

## 3.1 Introduction

Neural conversation models (NCMs), which learn a direct mapping between a dialogue history and a response utterance based on neural networks, are widely researched as an approach for building non-task-oriented dialogue systems [Vinyals and Le, 2015, Serban et al., 2016]. However, its response generation process is entirely black-boxed, unlike previous dialogue systems (e.g., modular-based dialogue systems), making it difficult to interpret why the model generates a response to a dialogue context. This problem complicates transferring the technology of NCMs to practical situations, such as introducing a chat-bot into business scenes. Introducing a mechanism for controlling the response generation of NCMs by clear intention helps to solve this problem.

Conventionally, dialogue act labels have been exploited to control the intention of response generation in both task- and non-task-oriented dialogue systems [Meguro et al., 2010, Yoshino and Kawahara, 2015, Shibata et al., 2016] . A dialogue act is defined as a unit that reveals the functions of associated turns in dialogues [Jurafsky, 1997]. Dialogue act labels are unique classes to distinguish the dialogue acts of given utterances (e.g., “Hello!!” can be abstracted to dialogue act label “Greeting”) [Boyer et al., 2010, Bunt et al., 2012b]. The dialogue acts clarify the roles of utterances in the dialogue context. In pragmatics, a speech act is also known as a similar classification standard to a dialogue act, an essential component to analyze and model both human-human and human-machine dialogues [Alston, 2000]. However, the benefits of dialogue act labels have not been exploited enough in NCMs, although studies of previous dialogue systems focus on modeling dialogues based on interpretable units, such as dialogue act labels. This inability limits the effectiveness of NCMs in several points. First, having interpretable system intentions by dialogue act labels enables humans to understand the behavior of dialogue systems. Second, modeling the high-level decision-making policy apart from response generation will enable the system to respond with a well-organized and thoughtful response. For example, as shown in 4.1, the required response to the system is often different even the recent dialogue context is the same. In such a case, the high-level decision by the dialogue manager is important, which uses other information such as user preference.

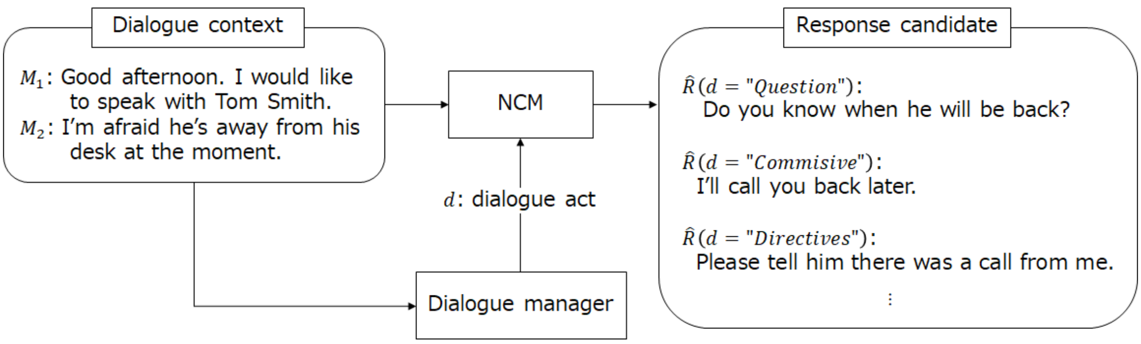



Figure 3.1: Response generation conditioned by dialogue act label: the NCM generates responses that represent dialogue act labels determined by a high-level decision-making model such as a dialogue manager.

Some existing studies have tackled this kind of problem for controlling responses of NCMs using actual labels (e.g., dialogue acts, emotions, persona); however, these works still suffer from limitations [Wen et al., 2015b, Li et al., 2016b, Huang et al., 2018b]. One crucial issue is that they do not have any explicit training objectives to guarantee that a generation has discriminability* to the given condition. In other words, to enforce the conditional generation of an

*Note that we use "discriminative" and "discriminability" as terms referring to whether a given dialogue act label is represented in response utterances.

NCM, we have to establish a new training objective which we call a label-aware objective.

To solve this problem, we introduce an extended framework of sequential generative adversarial networks [Yu et al., 2017] to improve the controllability of NCMs, given a dialogue act label as the condition. We propose a label-aware adversarial learning framework that alternatively trains both a generator, which creates a response to the given dialogue act label, and a discriminator, whose objective is to evaluate both the naturalness and the discriminability. The discriminator evaluates the validity of the generated responses by a classification model: label awareness. The evaluation results are used as a reward of reinforcement learning for training the generator. The method to control the response generation by adversarial learning is proposed by [Kawano et al., 2019]. We generalized the method from the viewpoint to control the generator by reinforcement learning.

In this study, we first describe the task of conditional response generation by dialogue act labels and existing approaches (Section 3.3). We introduce a reinforcement learning framework involving adversarial learning to address the problem of conditional response generation (Section 3.4). In experiments, we performed automatic and human evaluation of the controllability and naturalness of the generated responses (Section 3.5). The experimental results showed that our proposed model significantly improved the controllability scores in both automatic and human subjective evaluations, even it achieved better or comparable naturalness to existing methods (Section 3.6). We discuss the challenges for the advancement of conditional response generation given dialogue act labels in neural conversation models by analyzing our experimental results (Section 3.7).

## 3.2 Related work

Dialogue systems that have dialogue management modules determine a dialogue act or a frame of a system response using statistical methods, such as reinforcement learning [Young et al., 2010, Meguro et al., 2010, Yoshino and Kawahara, 2015, Keizer and Rieser, 2017]. Their response generation modules create responses according to these dialogue acts or frames based on rules, templates, agendas, or other statistical models [Oh and Rudnicky, 2000, Xu and Rudnicky, 2000]. In recent years, neural network-based generation modules have been widely used for building response generation modules [Dušek et al., 2018]. The controllability of neural generation systems is a big research issue.

One work proposed a conditional language model called Semantically Conditioned Long Short-Term Memory (SC-LSTM) [Wen et al., 2015b], which generates utterances based on dialogue act labels and frames in the domain of restaurant navigation dialogues using a gating mechanism. Furthermore, several improved SC-LSTM models are proposed in E2E NLG Challenge [Dušek et al., 2018]. However, their training framework requires state frames that entirely express the function and the contents of the target utterances. It is unrealistic to apply this method to building a non-task-oriented dialogue system. Another work proposed a neural conversation model (NCM) based on a conditional variational autoencoder [Zhao et al., 2017], which generates responses with high diversity at the discourse-level using latent variables considering dialogue acts. Since this model focuses on learning well-organized latent variables using dialogue act labels, its training objective insufficiently guarantees the discriminability of generated responses by dialogue act labels.

There is another research trend for controlling NCMs with a given condition, such as persona or emotion labels [Li et al., 2016b, Huang et al., 2018b]. These NCMs are optimized by softmax cross-entropy loss (SCE-loss), which calculates losses word-by-word. However, such existing training objectives do not necessarily guarantee that the generated response has high discriminability to the given class label. SCE-loss can optimize the word prediction itself, but it cannot evaluate whether the property of the generated responses belongs to the given classes. To address this problem, some works incorporate an auxiliary classifier to NCMs to improve the discriminability of their generated responses [Zhou and Wang, 2018, Shen and Feng, 2020]. However, their generation models must cope with over-fitting to the auxiliary classifier because the classifier is naively trained from the training data.

In this study, we introduce an extended framework of sequential generative adversarial networks [Yu et al., 2017, Li et al., 2017a, Tuan and Lee, 2019]. Different from methods incorporating with static auxiliary classifiers, our framework alternatively trains both a generator, which creates responses according to a given dialogue act label and a discriminator, which has an objective to evaluate both the naturalness and the discriminability of the generated responses by the given condition. Our architecture prevents the over-fitting problem to the auxiliary classifier by dynamically updating the discriminator through adversarial learning. Our architecture also makes it possible to consider the total quality of the generated responses, unlike SCE-loss, which is optimized for each term.

## 3.3 Conditional response generation by given dialogue act labels

In this study, we focus on controlling a conditional neural conversation model (NCM) using dialogue act labels. We assume a non-task-oriented dialogue as the target domain. In this section, we first address the conditional response generation task tackled in this study (Fig. 3.3.1). Then we describe the conventional architecture of a conditional response generation model based on minimizing softmax cross-entropy loss (Section 3.3.2).

### 3.3.1 Task settings

The task we focus on is building a controllable NCM by a given condition, typically a dialogue act label. The problem is defined as generating the response word sequence $\hat{R} = [\hat{w}_1, \hat{w}_2, \cdots, \hat{w}_{T'}]$ given dialogue history $M = [M_1, M_2, \ldots, M_N]$ and of response dialogue act label $d \in \mathcal{D}$. Here $N$ is the length of the dialogue history, $T'$ is the number of words in the response, and $\mathcal{D}$ is the set of dialogue act labels. As shown in 4.1, response $\hat{R}$ has to satisfy not only the behavioral characteristics of a given dialogue act label but also appropriateness in the dialogue context (=history).

We assume that a dialogue act label is decided by a high-level decision-making model, such as a dialogue manager, built separately from the NCM. This allows us to connect high-level decision-making models to the NCM freely, and we may enable to provide a useful generalization and a personalization to users. We only focus on the problem of whether NCMs can generate responses according to given dialogue act labels, since building the decision-making model is beyond the scope of this study. We use oracle dialogue act labels in a corpus when we evaluate response generation systems.

### 3.3.2 General conditional neural conversation model with dialogue act labels

We introduce a general conditional NCM controlled by given dialogue act labels as the baseline. We built a conditional NCM based on a hierarchical encoder-decoder model, which explicitly uses dialogue act labels in its decoding steps (Fig. 3.2). We adopt vector concatenation between a word vector and a vectorized condition to feed decoder input as a widely used method of conditional NCMs [Li et al., 2016b, Huang et al., 2018b].

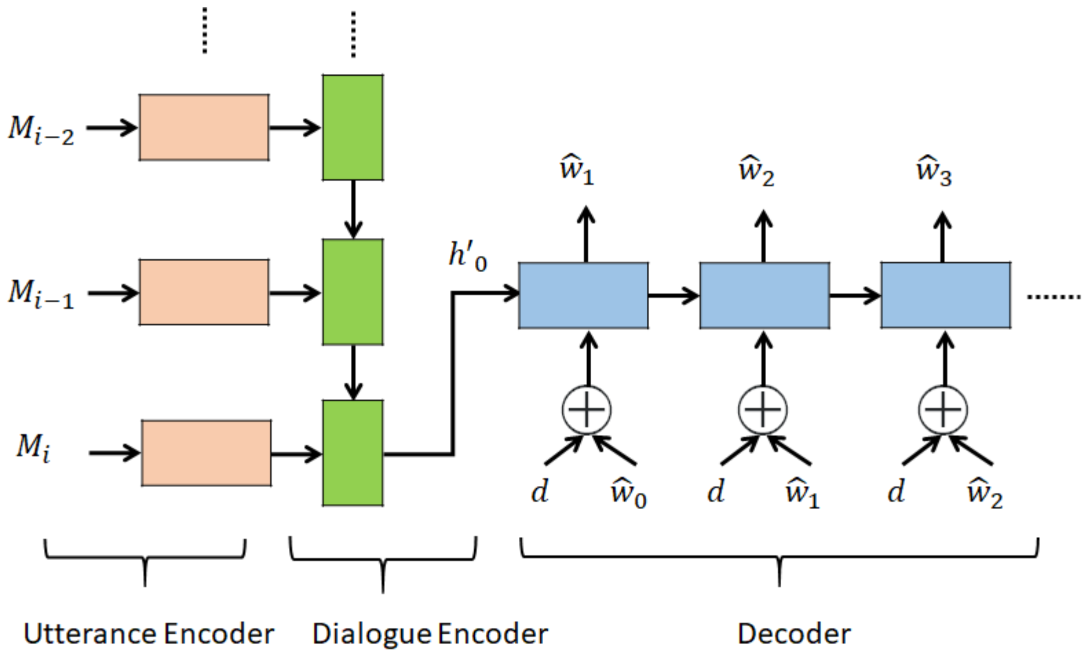



Figure 3.2: Conditional NCM with dialogue act labels

The encoder network has a hierarchical structure that consists of utterance and dialogue encoders [Serban et al., 2016]. The utterance encoder receives a word at each time step using forward RNNs to encode the utterance into a fixed-length vector:

$$h_{i,t} = \mathrm{RNN}_{\mathrm{utterance}}(h_{i,t-1}, \mathrm{Embed}(w_{i,t})). \tag{3.1}$$

Here $i$ is the number of turns in the dialogue context, and $h_{i,t}$ is the hidden vector obtained by inputting each word $w_{i,t}$ in utterance $M_i$. The embedding layer (Embed) projects $w_{i,t}$ to a fixed-length word vector to be used as input.

Utterance vectors are given to the dialogue encoder:

$$c_i = \mathrm{RNN}_{\mathrm{context}}(c_{i-1}, h_{i,T_i}). \tag{3.2}$$

Here $T_i$ is the number of words in the utterance $M_i$, and $h_{i,T_i}$ is a hidden vector obtained at the last step of the utterance encoder. Resultant vector $c_N$ is fed to the decoder and used as initial hidden state $h'_0$ to generate a response sentence. In the decoder, hidden state $h'_t$ of the decoder and the output probability of word $p_t$ are calculated:

$$h'_t = \mathrm{RNN}_{\mathrm{dec}}(h'_{t-1}, [\mathrm{Embed}(\hat{w}_{t-1}); \mathrm{Embed}_{\mathrm{da}}(d)]), \tag{3.3}$$

$$p_t = \mathrm{Softmax}(\mathrm{Linear}_{\mathrm{proj}}(h'_t)). \tag{3.4}$$

Here $\mathrm{Linear}_{\mathrm{proj}}$ is a projection layer, which maps $h'_t$ to a vector of vocabulary size $|\mathcal{V}|$. $\mathrm{Embed}_{\mathrm{da}}$ is a linear transformation layer that converts target dialogue act label $d$ into a fixed-length vector. $\hat{w}_t$ is sampled from a probabilistic distribution $p_t \in \mathbb{R}^{|\mathcal{V}|}$ obtained by Softmax function from the output of projection layer, and used as a part of the input of the next time step.

In this decoding architecture, we expect the decoding result to correspond to both the given dialogue act label and the dialogue history by inputting dialogue act label $d$ in addition to the already generated words. $d$ is encoded to a fixed-length vector using an embedding layer ($\mathrm{Embed}_{\mathrm{da}}$) as input.

In general, such NCMs are trained by minimizing the softmax cross-entropy loss (SCE-loss):

$$L_{\mathrm{SCE}} = -\sum_{t=1}^{T} \log p_t(w_t). \tag{3.5}$$

Here $T$ is the number of words in the ground-truth response $R = [w_1, w_2, \cdots, w_T]$, and $p_t(w_t)$ denotes the value that corresponds to the ground-truth word $w_t$ in the probabilistic distribution $p_t$ obtained from Softmax function from the output of projection layer.

SCE-loss optimizes the word prediction at each decoding step. However, it does not explicitly consider the given dialogue act label in the loss calculation during training. It only considers the likelihood of ground-truth words. Related to this, the NCM optimized by SCE often generates biased responses that contain frequently used words [Li et al., 2016a]. This problem will be critical in situations where the NCM must strictly represent a given dialogue act label. It is because dialogue acts of individual utterances change depending on dialogue context even if they have the same literal meaning (or contain frequent words representing the given dialogue act's characteristics). Due to such complexity of human language, SCE-Loss, which not considering the whole structure of the generated response, may not handle the differences in dialogue acts appropriately. In the following section, we address this problem by introducing an explicit training objective based on reinforcement learning involving adversarial learning for generating well-controlled responses by dialogue act labels.

## 3.4 Enhancing conditional response generation based on reinforcement learning

Conditional generation based on SCE-loss does not guarantee that the generation result follows a given condition. We introduce an objective function based on a framework of reinforcement learning (RL) that enhances the performance of the conditional response generation. This is because RL-based systems have the advantage of high flexibility in the design of reward functions, including the auxiliary classifier compared to other approaches [Hu et al., 2017]. Our RL-based system optimizes the conditional NCM by maximizing the reward, which evaluates whether the generated response obeys the given dialogue act label. In other words, we optimize the conditional NCM with guarantees that responses are generated according to a given dialogue act label. In the most current system, the auxiliary classifier is used to give rewards [Zhou and Wang, 2018]. However, since using a static classifier caused an over-fitting problem of the generator, we prevent this problem by introducing adversarial learning, which dynamically changes the classifier based on the generation results from the generator.

In this section, we explore different types of approaches for optimizing conditional NCMs. We first formulate a conventional optimization framework for a case with a static dialogue act classifier for RL and address its problem (Section 3.4.1 and Section 3.4.2). We extend this optimization framework based on the sequential generative adversarial network (SeqGAN) [Yu et al., 2017, Li et al., 2017a], including a different classifier and a dynamically changed classifier named a discriminator, for improving both the controllability and the naturalness of the conditional response generation (Section 3.4.3).

### 3.4.1 REINFORCE algorithm for conditional response generation

We introduce a policy gradient (REINFORCE Algorithm) [Williams, 1992], which is a direct differentiation of the reinforcement learning (RL) objective. The problem of response generation in NCMs is defined as generating response word sequence $\hat{R} = [\hat{w}_1, \hat{w}_2, \cdots, \hat{w}_{T'}]$ given dialogue context $M$. Such a word selection process is defined as a word-selecting action sequence, which is generated by an actual policy in a Markov decision process (MDP) [Ranzato et al., 2016, Li et al., 2017a]. We define a reward function based on the classifier to evaluate the validity of conditional response generation in NCMs. The evaluation score is fed as a reward to optimize the generator ' s policy by maximizing the expected reward of the generated responses. The gradient of the objective function is defined[†]:

$$\begin{aligned}\nabla J_{\mathrm{RL}}(\theta) \simeq & \frac{1}{T'} \sum_{t=1}^{T'} \sum_{\hat{w}_t \in \mathcal{V}} Q_{D_\phi}^{G_\theta}(\hat{R}_{1:t-1}, \hat{w}_t, d) \\ & \cdot \nabla_\theta G_\theta(\hat{w}_t | \hat{R}_{1:t-1}, d) \qquad (3.6) \\ = & \frac{1}{T'} \sum_{t=1}^{T'} \mathbb{E}_{\hat{w}_t \sim G_\theta}[Q_{D_\phi}^{G_\theta}(\hat{R}_{1:t-1}, \hat{w}_t, d) \\ & \cdot \nabla_\theta \log p_\theta(\hat{w}_t | \hat{R}_{1:t-1}, d)]. \qquad (3.7)\end{aligned}$$

Here $\theta$ is the parameters of the policy. $\mathcal{V}$ is a vocabulary, $\hat{R}_{1:t-1}$ indicates the al-

[†]Note that we followed the derivation and notations shown in [Yu et al., 2017].

ready generated word sequence, and $d$ is the dialogue act label used as a condition (state in MDP). $p_\theta(\hat{w}_t|\hat{R}_{1:t-1}) = G_\theta(\hat{w}_t|\hat{R}_{1:t-1}, d)$ is the generative probability of word $\hat{w}_t \in \mathcal{V}$ (action in MDP) in the decoder. $Q^{G_\theta}_{D_\phi}(\hat{R}_{1:t-1}, \hat{w}_t, d_t)$ is an action-value function that gives an expected future reward of word-generating action $\hat{w}_t$ given the state: already generated word sequence $\hat{R}_{1:t-1}$ and dialogue act label $d$. The expectation $\mathbb{E}[\cdot]$ can be approximated by sampling.

To evaluate the action-values for intermediate states, the Monte Carlo search under the policy of $G_\theta$ is applied to sample the future words. Each search ends until the end of word of response is sampled, or the sampled response reaches the maximum length. To obtain a stable reward and reduce the variance, we use an $N$-time Monte Carlo search[‡] [Yu et al., 2017] as:

$$\{\hat{R}^1_{1:T_1}, \cdots, \hat{R}^N_{1:T_N}\} = \mathrm{MC}^{G_\theta}(\hat{R}_{1:t}, d; N). \tag{3.8}$$

Here $T_n$ denotes the number of words in the response sampled by the $n$-th Monte Carlo search. $(\hat{R}_{1:t}, d)$ is the current state and $\{\hat{R}^1_{1:T_1}, \cdots, \hat{R}^N_{1:T_N}\}$ are sampled from the policy $G_\theta$. The reward function provides $N$ rewards for the sampled $N$ responses, respectively. The final reward for the intermediate state is calculated as the average of the $N$ rewards. Therefore, we calculate the reward in time-step $t$ for the generated response with the length $T'$, as:

$$Q^{G_\theta}_{D_\phi}(\hat{R}_{1:t-1}, \hat{w}_t, d) = \begin{cases} \frac{1}{N}\sum_{n=1}^{N} D_\phi(\hat{R}^n_{1:T_n}, M, d) & \text{for} \quad t < T' \\ D_\phi(\hat{R}_{1:T}, M, d) & \text{for} \quad t = T'. \end{cases} \tag{3.9}$$

Here $D_\phi(\cdot)$ is a reward function given by a classifier, which has a parameter $\phi$. The classifier evaluates the validity of the generated responses to the given condition $d$. We introduce different classifiers to update the response generation model.

[‡]We set $N = 5$. However, since the computation cost of the Monte Carlo search is high when training the large model, we can also adopt an approach for speeding up the training, such as REGS [Li et al., 2017a].

### 3.4.2 Static reward function based on dialogue act classifier

We first introduce a naive reward function using a static dialogue act classifier. This is inspired by previous works that introduced an auxiliary classifier to encourage conditional generation [Zhou and Wang, 2018,Shen and Feng, 2020]. The auxiliary classifier predicts whether the generated responses follow the given condition with a multi-class classifier. In other words, it forces NCM ' s response generation to follow a given condition, such as the dialogue act label. We built classifier $D_\phi$ based on a hierarchical encoder to predict the dialogue act label of generated response $\hat{R}$ (Fig. 3.3).

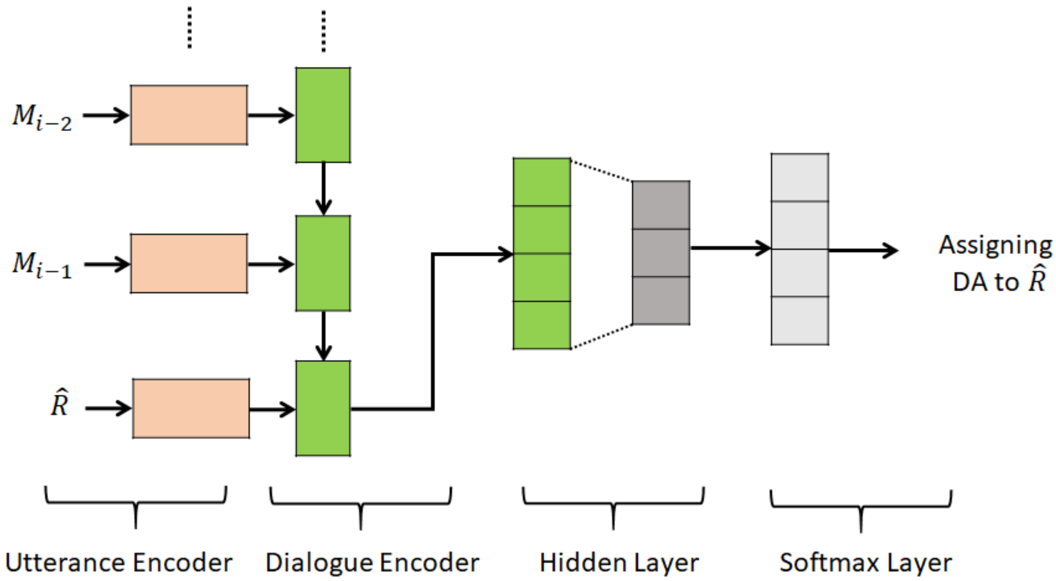



Figure 3.3: Dialogue act classifier with hierarchical encoder

Dialogue history $M$ and generated responses $\hat{R}$ are encoded to contextual representation $c$ by the hierarchical encoder. We use the same encoder structure of the conditional NCM formulated in Eqs. (3.1)-(3.2) for encoding the dialogue history and the generated response. Contextual representation $c$ is fed into a fully connected layer to classify the dialogue act of response $\hat{R}$:

$$h_{\mathrm{da}} = \mathrm{MLP}(c), \tag{3.10}$$

$$p_{\mathrm{da}} = \mathrm{Softmax}(\mathrm{Linear}_{\mathrm{da}}(h_{\mathrm{da}})). \tag{3.11}$$

Here MLP is a multi-layer perceptron that applies the Relu function to each hidden layer, and $\mathrm{Linear}_{\mathrm{da}}$ is a linear transformation layer, which transforms $h_{\mathrm{da}}$ to a vector of a number of unique dialogue act labels. $p_{\mathrm{da}}$ is the classification probability of the dialogue act of $\hat{R}$ to class $d$.

We use posterior probability $p_{\mathrm{da}}(d|\hat{R}, M)$ estimated by the classifier as the reward of the generator. It gives higher rewards if the posterior probability to given dialogue class $d$ is high. We expect that this classifier will encourage training the generator to make controlled responses according to the given dialogue act labels. However, it may cause a problem where the generator is over-fitted to the dialogue act classifier; when using a classifier that is naively trained by maximum likelihood estimation, the conditional NCM invariably learns a "lazy" policy and chooses the easiest way to represent a given dialogue act. This choice will improve the controllability, but it will not lead to natural responses. To prevent such problems, optimization based on SCE-loss as pre-training must be incorporated.

### 3.4.3 SeqGAN for conditional response generation with dialogue acts

We introduce the extended framework of the sequential generative adversarial network (SeqGAN) [Yu et al., 2017]. SeqGAN uses a discriminator instead of a classifier, which dynamically updates the evaluation system of the generation results. Our framework alternatively trains both the generator, which generates a response according to a given dialogue act label, and the discriminator, which has an objective that evaluates both the naturalness and the discriminability of a generated response for a given dialogue act label. A benefit of such a training strategy is that the system can dynamically update the parameters for further iteratively improving the generative model while avoiding any over-fitting to evaluation system; discriminator $D_{\phi}$. It can penalize a response that is unnatural or typical, even if the generated response represents the given dialogue act label. [Kawano et al., 2019] proposed this system to use adversarial learning for controlling the generator. We regard the system as an extension of a framework based on a static reward function described in the previous section.

In our framework, we incorporated dialogue act labels in the discriminator in a general SeqGAN (Fig. 3.4). We propose two discriminators for incorporating dialogue act label information in the discriminator: implicit and explicit. We also propose an ensemble approach that integrates both discriminators. Each method is described below in respective sections.

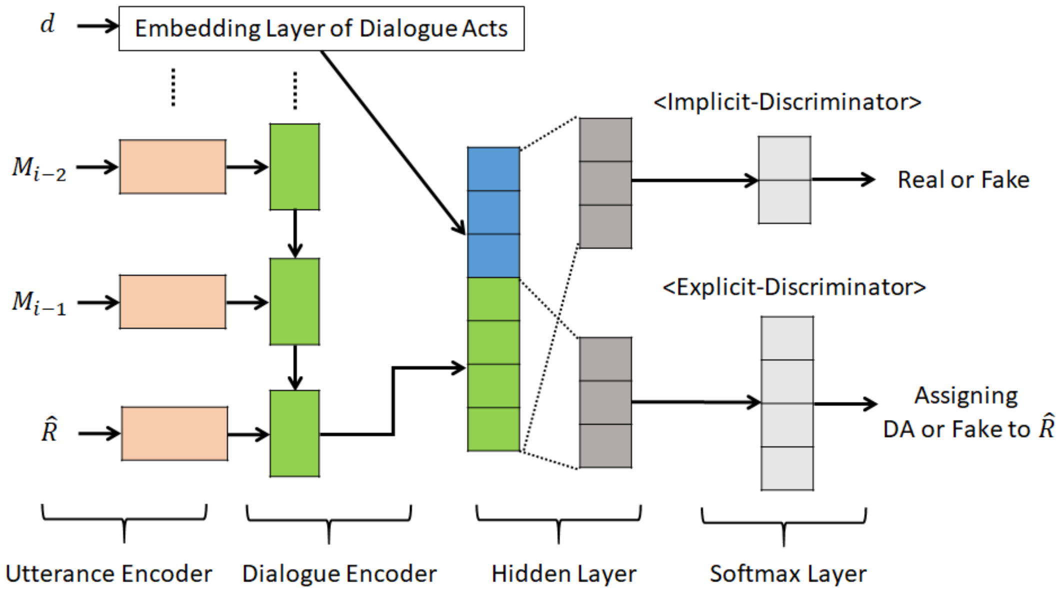



Figure 3.4: Implicit & explicit-discriminator

### Binary objective: implicit-discriminator

We first propose an implementation of the discriminator that incorporates dialogue acts in its additional feature vectors. We use the same structure of the dialogue act classifier described in Section 3.4.2 and apply a concatenated vector between contextual representation $c$ and the embedding vector of the dialogue act label as the MLP input. We also change the dimension of the output layer to 2. We call this architecture "implicit" and define its objective function:

$$\min_{\phi} -\mathbb{E}_{R\sim p_{\mathrm{data}}(\cdot|M,d)}[\log D_\phi(R,M,d)]$$
$$-\mathbb{E}_{\hat{R}\sim G_\theta(\cdot|M,d)}[\log(1-D_\phi(\hat{R},M,d))]. \tag{3.12}$$

Here $p_{\mathrm{data}}$ is the probability distribution of actual response $R$ on the training data. $D_\phi(\hat{R},M,d)$ is the probability of response $\hat{R}$ belonging to the binary classes: an actual response in the training data (real) or a generated response (fake).

We expect that the implicit discriminator will use the information of the dialogue acts as a feature and discriminate the generated results as fakes if they do not follow a given dialogue act label (Fig. 3.4, upper-right). Some works have similar approaches in emotional response generation [Sun et al., 2018,Kong et al., 2019]. However, this discriminator remains a simple extension of the standard discriminator, which classifies responses in two classes. In other words, since the objective is not changed, it probably struggles to distinguish the class (dialogue act label) of the responses. We introduce another discriminator to solve this problem in the next section.

#### Multi-class objective: explicit-discriminator

We propose an approach that extends the classification problem of the discriminator from the binary classification of fake/real to multi-class classification that distinguishes target dialogue act classes (Fig. 3.4 lower-right). It also applies an additional class to the static dialogue act classifier. This discriminator has a multi-class objective for $|\mathcal{D}|+1$ class classification. Here $|\mathcal{D}|$ is the number of unique dialogue act labels; another one is a fake class for categorizing the responses as generated. We call this architecture "explicit" and define its objective function:

$$\min_{\phi} -\sum_{i=1}^{|\mathcal{D}|}\mathbb{E}_{R\sim p_{data}(\cdot|M,d)}[\log D_\phi(d=i|R,M)]$$
$$-\mathbb{E}_{\hat{R}\sim G_\theta(\cdot|M,d)}[\log D_\phi(d_{\mathrm{fake}}|\hat{R},M)]. \tag{3.13}$$

We used posterior probability $D_\phi(d|\hat{R},M)$ estimated by the discriminator as the generator ' s reward. We expect that this discriminator will encourage generator training to provide discriminative sentences with dialogue acts because the generation results are penalized if they are not natural and fail to follow the manners of the given dialogue acts. Similar approaches, which use multi-class objectives in GAN, have been proposed for image generation [Odena, 2016, Salimans et al., 2016]. These works suggested that multi-class objectives are useful for stable training on a small amount of training data.

#### Ensemble objective: ensemble-discriminator

[Kawano et al., 2019] indicated that the implicit-discriminator generally focuses more strictly on classifying real or fake, whereas the explicit-discriminator tends to focus on classifying dialogue act labels. However, the explicit-discriminator ' s properties resemble those of the static dialogue act classifier if the training is over-fitted. The explicit discriminator will cause the same over-fitting problem to the classifier, if it always provides a fixed amount of fake probability.

Thus, we propose an ensemble approach that combines the properties of both the implicit-discriminator and the explicit-discriminator ($D_{\phi^{\mathrm{imp}}}$ and $D_{\phi^{\mathrm{exp}}}$) for stable training. We define the reward function using the harmonic-mean:

$$D_{\phi^{\mathrm{imp}},\phi^{\mathrm{exp}}}(\hat{R}, M, d) = 2 \cdot \frac{D_{\phi^{\mathrm{imp}}}(\hat{R}, M, d) \cdot D_{\phi^{\mathrm{exp}}}(d|\hat{R}, M)}{D_{\phi^{\mathrm{imp}}}(\hat{R}, M, d) + D_{\phi^{\mathrm{exp}}}(d|\hat{R}, M)}. \tag{3.14}$$

This reward function $D_{\phi^{\mathrm{imp}},\phi^{\mathrm{exp}}}$ simultaneously optimizes both (3.12)-(3.13). This is expected to ensure that the NCM does not learn the biased policy by the characteristics of one or the other discriminator. By introducing such a harmonic mean, unlike a simple additive mean, a higher reward can be given when the scores of both discriminators are in a high state.

#### Training procedure

The training procedure of a conditional NCM with SeqGAN is shown in Algorithm 1.

First, we pre-train the conditional NCM by minimizing the SCE-loss ($L_{\mathrm{SCE}}$). Then we apply alternate training between the generator and the discriminator. In G-steps for updating the generator, we sample generated response $\hat{R}$ with

**Algorithm 1** Training procedure

**Require:** Generator policy $G_\theta$; roll-out policy $G'_\theta$; discriminator $D_\phi$

Initialize $G_\theta$ with random weights $\theta$
Pretrain $G_\theta$ to minimize $L_{\text{SCE}}$
**for** Number of iterations **do**
    $G'_\theta \leftarrow G_\theta$
    **for** Number of G-steps **do**
        Sample $(M, R, d)$ from training data
        Generate response $\hat{R}$ using $G'_\theta$ on $(M, d)$
        Compute $Q^{G_\theta}_{D_\phi}$ for $(M, \hat{R}, d)$ using $G'_\theta$
        Update $G_\theta$ based on $\nabla J_{\text{RL}}(\theta)$
    **end for**
    **for** Number of D-steps **do**
        Sample $(M, R, d)$ from training data
        Generate response $\hat{R}$ using $G_\theta$ on $(M, d)$
        Update $D_\phi$ using $(M, \hat{R}, d)$ and $(M, R, d)$
    **end for**
**end for**

dialogue history $M$ and dialogue act label $d$ then calculate $Q^{G_\theta}_{D_\phi}$ by discriminator $D_\phi$. We update the parameters of $G_\theta$ using the calculated $Q^{G_\theta}_{D_\phi}$. Here we perform the update by $L_{\text{CE}}$ after updating by RL to stabilize the training. This approach works as teacher-forcing and prevents the collapse of policies due to the model ' s inability to access the reference response [Li et al., 2017a]. In D-steps for updating the discriminator, we sample real response $R$, given dialogue history $M$ and dialogue act label $d$ from the training data, and then sample fake response $\hat{R}$ from generator $G_\theta$. We update the parameters of discriminator $D_\phi$ using the real and fake samples. Note that when we apply the naive RL-based method presented in Section 3.4.2, the system does not use D-steps. Instead, a pre-trained classifier of dialogue act classes is used as the discriminator.

## 3.5 Experimental settings

We experimentally investigated the advantages of the proposed controlled response generation system of both its controllability and naturalness. We performed both automatic and human evaluations. In this section, we describe their experimental settings.

### 3.5.1 Dataset

We used the DailyDialog corpus [Li et al., 2017b] and the Switchboard dialogue act corpus [Jurafsky, 1997] for the evaluations. The DailyDialog corpus consists of ten categories from a wide variety of topics. We used training/validation/test sets of the DailyDialog (Table 3.1). This corpus is annotated with four dialogue act labels for each utterance: inform, questions, directives, and commissive.

Table 3.1: Number of dialogues/utterances in DailyDialog

| | Dialogues | Utterances |
|---|---|---|
| Train | 11,118 | 76,052 |
| Validation | 1,000 | 7,069 |
| Test | 1,000 | 7,740 |

The switchboard dialogue act corpus (SWBD), which is a large-scale corpus containing telephone speech [Stolcke et al., 2000, Jurafsky, 1997], is annotated with the SWBD-DAMSL tag-set, which consists of 42 dialogue act labels (Table 3.2). We also used this dataset for the evaluations. Since SWBD is a conversation with more than one utterance per turn, unlike the DailyDialog, we added to the suffix special symbols that indicate the type of conversation floor in each utterance [Zhao et al., 2017].

Table 3.2: Number of dialogues/utterances in SWBD

| | Dialogues | Utterances |
|---|---|---|
| Train | 2,316 | 20,788 |
| Validation | 60 | 5,255 |
| Test | 62 | 5,481 |

### 3.5.2 Competing models

We compare the different kind of training objectives and neural conversation models.

### Model settings

Based on a hierarchical encoder, we used the following types of NCMs as baselines:

- **ASEQ2SEQ**: a standard neural conversation model that encodes a previous utterance as a query for decoding a response with an attention mechanism (general-attention) [Luong et al., 2015].
- **HED**: a hierarchical encoder-decoder model [Serban et al., 2016] without conditioning to a decoder described in Section 3.3.2.
- **CHED**: a HED model with a conditioning mechanism. We gave the condition (dialogue act labels) described in Section 3.3.2.

We can also use other advanced networks for neural conversation generation [Zhou et al., 2018, Peng et al., 2019]. However, note that we focus on training objectives to enhance the generation performance of the NCMs.

### Training objectives

We trained these NCMs with the following training objectives:

- **w/ SCE**: optimization by minimizing the softmax cross-entropy loss (SCE-loss) described in Section 3.3.2.
- **w/ KgCVAE**: optimization by knowledge-guided conditional variational autoencoder (KgCVAE) [Zhao et al., 2017]. We used CHED for the basic network, and we did not use any meta-features except for dialogue acts. The generation performance depends "heavily" on randomness due to sampling from a latent variable, thus we calculate the average results of five trials when we input an target dialogue act.
- **w/ AC**: joint optimization with auxiliary classifier. Unlike RL-based approaches, we use sampling from a multinomial distribution parametrized by softmax function for receiving the feed-back from auxiliary classifier. We basically follow the algorithm described in [Hu et al., 2017]. We use CHED as the generator network, and the dialogue act classifier described in Section 3.4.2 as the auxiliary classifier. Here, we removed objective function

and network related to VAE according to their official implementation[§], since the training of latent variable is not the scope of this study.

- **w/ RL**: optimization by the static reward function based on the dialogue act classifier described in Section 3.4.2.
- **w/ IMPLICIT**: optimization by SeqGAN with the implicit-discriminator described in Section 3.4.3.
- **w/ EXPLICIT**: optimization by SeqGAN with the implicit-discriminator described in Section 3.4.3.
- **w/ ENSEMBLE**: optimization by SeqGAN with the implicit and explicit-discriminator described in Section 3.4.3.

**Training settings**

We used the same hyper-parameter settings in these NCMs and classifiers. The vocabulary size was 15000, the word embedding size was 300, the dialogue act embedding size was 50, and the hidden vector size was 300. We used a two-layer Gated Recurrent Unit (GRU) [Cho et al., 2014] as an RNN. In training, we used a mini-batch size of 64, a G-step of 1, a D-step of 5, and an Adam optimizer [Kingma and Ba, 2014] with a learning rate of 1e-5. We used teacher-forcing in the training process [Vinyals and Le, 2015]. We used the model with the highest controllability for the generated response and the deterioration of the perplexity within 1.0 points from the pre-training model on the validation set.

### 3.5.3 Automatic evaluation metrics

We automatically evaluated the generation results using references in the test set. We used a beam search (a beam-width of 5) for generating examples for evaluation [¶]. For automatic evaluations, we used the following seven metrics:

- **Perplexity** (PPL) is a general metric for evaluating a language model performance. The model likelihoods of the reference responses are calculated.

[§]https://github.com/asyml/texar/tree/master/examples/text_style_transfer
[¶]We used the top-1 response from the beam search candidates.

Note that the perplexity scores do not directly reflect the generation quality; for example, dull responses also have good perplexity scores.

- **ROUGE**, which is a popular automatic evaluation metric of language generation tasks, calculates the similarity between references and generated responses [Lin and Och, 2004] based on n-gram recall. We used ROUGE-L, which is a variant of ROUGE. ROUGE-L compares the references and the generated responses based on the longest common sub-sequences between them.

- **Vector Pooling** calculates the cosine similarity between the reference and generated response vectors [Tao et al., 2017]. Each sentence vector is calculated by concatenating both the max and the min-pooling vectors of the word vectors in a sentence. This is a simple extension of vector extrema [Liu et al., 2016], which is widely used in dialogue evaluations.

- **NN Scorer** is a neural network-based scorer that measures the relatedness between the generated response and its dialogue context [Tao et al., 2017]. The scorer is trained by the negative samples of response pairs and randomly selected contexts as well as the positive samples of the real pairs of a response and its context. Note that it does not refer to ground-truth responses during the evaluations.

- **RUBER** is a blending metric between the referenced (Vector Pooling) and the unreferenced metric (NN Scorer) [Tao et al., 2017]. It outperforms the embedding-based and word-overlap-based metrics in many cases [Tao et al., 2017]. We followed the setting in a previous work [Tao et al., 2017] for training the NN scorer and inferring the score. We used both DailyDialog and SWBD to train the model.

- **Entropy** (Ent) is a diversity metric [Zhang et al., 2018] that reflects the evenness of the empirical n-gram distribution for the given responses:
  $Ent = \frac{1}{\sum_{w \in \mathcal{V}} C(w)} \sum_{w \in \mathcal{V}} C(w) \log \frac{C(w)}{\sum_{w' \in \mathcal{V}} C(w')}$,
  where $\mathcal{V}$ is a set of all n-grams in the given responses and $C(w)$ denotes the frequency of n-gram $w$. We set the 4-gram for evaluation.

- **Controllability** is the classification accuracy of the pre-trained dialogue act classifier determined using the training set[‖] described in Section 3.4.2. We used the same dialogue act classifier. Any generated sentences are labeled by the classifier and compared with the given condition label to calculate the precision, recall, and f1.

### 3.5.4 Human subjective evaluation metrics

Since automatic evaluation scores suffer from a lack of correlation with human subjective evaluation results [Liu et al., 2016], We also examined models with a human subjective evaluation to confirm the naturalness and controllability of the generated responses on DailyDialog.

#### Naturalness evaluation

We used a 3-point scale in accordance with an existing work [Xing et al., 2017] and randomly selected 240 generated responses from the test set. Human annotators added their evaluation scores for each sample by looking at the dialogue contexts. A detailed description follows:

- **+2**: This response is not only relevant and natural but also informative and interesting.
- **+1**: This response can be used as a response in the context, although it is universal, like "Yes, I see," "Me too," or "I don ' t know."
- **0**: This response cannot be used in this context. It is either semantically irrelevant or disfluent.

Three annotators evaluated each sample, and their final score was decided by a simple majority. If all three evaluators completely disagreed (0, +1, and +2), the example was scored as 1.

[‖]The accuracy of the classifiers was 81.59% (DailyDialog) and 87.27% (SWBD) in the test sets.

### Controllability evaluation

We manually evaluated whether the generated responses follow the given dialogue act labels. An expert annotator with three years of experience in dialogue act annotation categorized the dialogue acts for the generated responses. The annotator was trained using the training data of the DailyDialog corpus before the evaluation.

Table 3.3: Automatic evaluation results for each neural conversation model. Coherence and controllability scores are displayed on a scale from 0 to 100.

| | | | | Coherence | | | | Controllability | |
|---|---|---|---|---|---|---|---|---|---|
| Dataset | Models | PPL | Ent | ROUGE-L | Vector Pooling | NN Scorer | RUBER | Prec. | Rec. |
| DailyDialog | ASEQ2SEQ | 34.97 | 10.90 | 15.31 | 84.04 | 46.46 | 42.52 | 48.41 | 50.98 |
| | HED | 33.03 | 9.59 | 13.79 | 83.87 | 47.12 | 43.08 | 46.90 | 48.83 |
| | CHED w/ SCE | 32.11 | 10.16 | 16.66 | 83.94 | 46.11 | 42.40 | 86.65 | 86.01 |
| | CHED w/ KgCVAE | 31.56 | 9.57 | 15.49 | **84.47** | 38.64 | 35.60 | 86.31 | 85.50 |
| | CHED w/ AC | 32.74 | 10.07 | 17.03 | 83.97 | 47.03 | 43.09 | 87.72 | 87.86 |
| | CHED w/ RL | 32.52 | 9.10 | **18.32** | 84.44 | 47.81 | 43.96 | **94.22** | **94.38** |
| | CHED w/ IMPLICIT | 32.12 | 10.09 | 17.90 | 84.09 | **48.52** | **44.45** | 86.77 | 86.20 |
| | CHED w/ EXPLICIT | 32.27 | 10.14 | 17.76 | 83.51 | 47.94 | 43.89 | 89.17 | 89.18 |
| | CHED w/ ENSEMBLE | 32.30 | 10.31 | 17.98 | 83.53 | 48.39 | 44.32 | 90.51 | 90.52 |
| SWBD | ASEQ2SEQ | 44.86 | 2.97 | 14.01 | 57.67 | 51.61 | 42.10 | 26.69 | 22.80 |
| | HED | 44.30 | 3.32 | 16.17 | 59.18 | 54.86 | 44.17 | 46.62 | 31.31 |
| | CHED w/ SCE | 41.18 | 4.92 | 26.20 | 71.78 | 51.22 | 45.59 | 92.14 | 74.82 |
| | CHED w/ KgCVAE | 42.22 | 3.42 | 25.47 | 69.14 | 53.46 | 47.59 | 92.58 | 65.63 |
| | CHED w/ AC | 44.05 | 4.81 | 25.22 | 70.33 | 54.77 | 48.01 | 93.87 | 74.95 |
| | CHED w/ RL | 41.37 | 3.40 | 26.82 | 74.18 | 53.66 | 47.44 | **96.78** | **97.41** |
| | CHED w/ IMPLICIT | 41.22 | 6.67 | 26.55 | 72.76 | 53.09 | 46.95 | 89.50 | 79.60 |
| | CHED w/ EXPLICIT | 41.27 | 6.75 | **27.08** | **74.51** | 61.86 | 54.03 | 93.55 | 94.13 |
| | CHED w/ ENSEMBLE | 41.29 | 5.70 | 26.97 | 74.21 | **62.41** | **54.46** | 95.20 | 95.79 |

## 3.6 Experimental results

### 3.6.1 Automatic evaluation results

#### Overall results

Table 3.3 shows the results of the automatic evaluation. We compared our proposed models based on SeqGANs ( w/ IMPLICIT, w/ EXPLICIT, w/ ENSEMBLE) with the baseline models ( w/ SCE, w/ KgCVAE, w/ AC, w/ RL). $\overline{\text{Prec.}}$ and $\overline{\text{Rec.}}$ are the weighted averages** of the precision/recall of each label of dialogue act.

For coherence, we confirmed that the models conditioned by the dialogue act labels (w/ SCE, w/ AC, w/ RL, w/ IMPLICIT, w/ EXPLICIT, w/ ENSEMBLE) improved the coherence scores (ROUGE-L, NN Scorer, and RUBER) compared to the models without conditions. In addition, the CHED models based on RL and SeqGANs achieved consistent improvement for each score compared with CHED w/ SCE. This result suggests that dialogue act conditions are a potential training constraint to improve the quality of the generated responses.

For controllability, CHED models based on RL and SeqGANs showed the improvement in controllability scores, although PPL was not worse than CHED w/ SCE. In particular, CHED w/ RL and CHED w/ ENSEMBLE showed the highest and second-highest improvement (over 90% $\overline{\text{Prec.}}$ and $\overline{\text{Rec.}}$). This advantage is more critical when using other decoding methods (e.g., random sampling and diverse beam search [Ippolito et al., 2018]). On the other hand, CHED w/ IMPLICIT ’ s improvement was limited by a lack of explicit objectives to identify dialogue acts, unlike w/ RL, w/ EXPLICIT, and w/ ENSEMBLE. CHED w/ KgCVAE did not show the improvement of controllability than CHED w/ SCE. This result is probably caused by the generation process of KgCVAE, which strongly depends on sampling from latent variables. In other words, depending on the state of the hidden vectors sampled from the encoder network to generate the response, the discriminability of that generated response can vary greatly. CHED w/ AC showed higher controllability than CHED w/ SCE; however, the controllability was less than RL and SeqGAN-based models. This is probably

**we use frequencies of each dialogue act label as weights.

because of the sparsity of error signals from the discriminator when both the discriminator and generator over-fit the training data. Note that discriminability of CHED w/ KgCVAE and w/ AC, can be enhanced by applying our SeqGAN-based methods to decoder networks in fine-tuning after their training.

For diversity, the CHED models based on SeqGANs showed comparable or slight improvement compared to CHED w/ SCE. However, CHED w/ RL did not improve diversity, which especially decreased on SWBD. This suggests that CHED w/ RL generated responses that contain specific vocabulary or typical phrases.

### Controllability results

We evaluated the case where the model was always given a fixed dialogue act as a condition, not only oracle. Table 3.6 shows the controllability score, diversity score, and NNScorer (unrefereed metric) score calculated for responses, which is generated by using all combinations of dialogue context and dialogue acts in the test set. Here, the table also includes results for cases where low-frequency dialogue act labels were excluded to eliminate the influence of dialogue act labels that rarely appear in the test set. For controllability, the model w/ RL showed the best performance. However, while the model w/ RL has achieved the desired dialog acts with high accuracy, the diversity of its responses has shown the lowest results compared to the models based on SeqGAN. The models w/ EXPLICIT and w/ ENSEMBLE have better controllability overall, other than the model w/ RL. Furthermore, when we evaluated the model by focusing only on the frequent dialogue acts, we confirmed that the models w/ RL, w/ EXPLICIT, and w/ ENSEMBLE more greatly improved the controllability than the model w/ SCE. The results of the NN Scorer showed comparable performance for all models.

For a detailed analysis of the controllability of the response generation, we show the controllability score of CHED w/ ENSEMBLE for each oracle dialogue act label in Table 3.5 (DailyDialog) and Table 3.6 (SWBD). The tables show the precision, recall, and the harmonic mean (F1) of each dialogue act and the improvement from the score of CHED w/ SCE, which achieved the best controllability in the baselines (**improv.**).

Our proposed CHED w/ ENSEMBLE improved the controllability of most classes, even if the target dialogue act labels have similar characteristics (e.g., "Directives" and "Commissive"). CHED w/ RL and CHED w/ EXPLICIT also showed a very similar improvement trend in the scores. However, we still must improve the controlled response generation based on minority dialogue act labels. We should not strongly accept the evaluation results from the dialogue act classifier, because annotating dialogue acts requires a great deal of expertise. Thus, a human expert annotator also evaluated the controllability of each model.

We should not strongly accept the evaluation results from the dialogue act classifier, because annotating dialogue acts requires a great deal of expertise. Thus, a human expert annotator also evaluated the controllability of each model in the next section.

### 3.6.2 Results of human subjective evaluation

Table 3.7 and Table 3.8 show the human evaluation results for DailyDialog ' s naturalness and controllability. Regarding the naturalness of the generated responses (Table 3.7), models based on SeqGANs generated more acceptable responses to the dialogue context. In particular, CHED w/ IMPLICIT achieved the highest performance, followed by CHED w/ ENSEMBLE. In contrast, HED w/ RL showed a slight decrease in overall naturalness despite being comparable to models based on SeqGANs in the automatic evaluations.

Regarding the controllability of the generated responses (Table 3.8), CHED w/ EXPLICIT and CHED w/ ENSEMBLE achieved the highest and second-highest performances. CHED w/ RL showed slightly inferior or comparable performance to CHED w/ EXPLICIT and CHED w/ ENSEMBLE, although CHED w/ RL remarkably improved controllability compared with the other models in the automatic evaluation. This is because the evaluation result of the dialogue act classifier used as a reward function on RL does not always agree with human evaluations. In other words, the CHED w/ RL model may be over-fitted to the dialogue act classifier. It learns strategies that is over-fitted to the dialogue act classifier, and the naturalness of the generated responses is only considered by the SCE-loss used as an auxiliary objective. In contrast, our models using the explicit-discriminator, which is dynamically updated by adversarial learning, have the potential to generate more natural and discriminable responses.

These results suggest that the implicit-discriminator improves the naturalness, and the explicit-discriminator improves the controllability in our adversarial learning. Furthermore, the ensemble approach, which combines both discriminators, has the potential to produce better naturalness and controllability results.

### 3.6.3 Dialogue examples

We show examples generated by our models in Table 4.7. Our proposed CHED w/ ENSEMBLE generated a comparable or slightly natural response compared to other models, even though their responses represent the characteristics of dialogue act labels when given during response generation. Here, note that the output of the listed models is very similar in some cases (e.g., first and second examples). This is because CHED w/ SCE is shared as the pre-trained model for each model. Hence, if the possible responses that CHED w/ SCE might generate for a given context are already appropriate, or if enough search in RL is not performed, the response generation's tendency may not vary significantly in case.

## 3.7 Conclusion

We introduced an extended framework of sequential generative adversarial networks to improve the controllability of neural conversation models (NCMs) under the constraint of a given dialogue act label. We propose a different kind of adversarial approach, including a new label-aware objective to encourage the generation of discriminative responses by given dialogue act labels. Experimental results showed that our proposed method, which is based on an ensemble rewarding strategy, has higher controllability for dialogue acts even though it has higher or comparable naturalness to existing methods.

However, the response quality of our proposal still has room for improvement. According to the human evaluation results, 70% of the generated responses did not cause dialogue breakdowns. However, their responses remain inadequate to satisfy the user. It is difficult to continue multi-turn conversations due to the limitation of the training data in the available dataset, such as DailyDialog and SWBD corpus. Thus, we need to explore an approach to address this problem.

Exploring a method to control the response of NCMs based on BERT [Devlin et al., 2019] or DialoGPT [Zhang et al., 2020], which are pre-trained on massive dialogue corpora, is one promising direction. Furthermore, we will integrate the optimization of natural language understanding and dialogue management into our framework while focusing on different types of labels and intentions (e.g., emotions, persona, convergence and divergence, etc.) for building a more user-satisfied conditional NCM.

Table 3.4: Automatic evaluation results of response generation on all combination of dialogue act and dialogue context in test set.

| | Models | $\overline{\text{Prec.}}$ | $\overline{\text{Rec.}}$ | Ent | NN Scorer |
|---|---|---|---|---|---|
| DailyDialog | CHED w/ SCE | 72.74 | 68.55 | 10.41 | 44.40 |
| | CHED w/ KgCVAE | 72.05 | 66.13 | 9.56 | 36.07 |
| | CHED w/ AC | 74.54 | 70.17 | 10.42 | 45.92 |
| | CHED w/ RL | 84.67 | 82.25 | 9.54 | 46.69 |
| | CHED w/ IMPLICIT | 72.85 | 66.87 | 10.37 | 47.80 |
| | CHED w/ EXPLICIT | 76.38 | 72.20 | 10.49 | 46.61 |
| | CHED w/ ENSEMBLE | 78.45 | 74.70 | 10.45 | 46.67 |
| SWBD | CHED w/ SCE | 34.78 | 31.18 | 5.89 | 48.95 |
| | CHED w/ KgCVAE | 36.58 | 33.78 | 6.06 | 47.18 |
| | CHED w/ AC | 26.76 | 31.11 | 4.28 | 48.53 |
| | CHED w/ RL | 33.68 | 33.92 | 4.68 | 51.92 |
| | CHED w/ IMPLICIT | 32.58 | 31.19 | 7.56 | 49.71 |
| | CHED w/ EXPLICIT | 32.51 | 33.34 | 8.80 | 50.63 |
| | CHED w/ ENSEMBLE | 33.52 | 33.10 | 7.23 | 50.01 |
| SWBD (freq. of DA $>$ 5) | CHED w/ SCE | 59.48 | 49.22 | 6.52 | 50.47 |
| | CHED w/ KgCVAE | 61.41 | 50.37 | 5.79 | 48.95 |
| | CHED w/ AC | 57.91 | 52.23 | 3.81 | 47.11 |
| | CHED w/ RL | 59.57 | 54.63 | 5.46 | 51.63 |
| | CHED w/ IMPLICIT | 59.21 | 49.33 | 7.82 | 51.14 |
| | CHED w/ EXPLICIT | 57.31 | 52.94 | 8.53 | 49.53 |
| | CHED w/ ENSEMBLE | 58.64 | 52.87 | 7.41 | 50.57 |
| SWBD (freq. of DA $>$ 20) | CHED w/ SCE | 79.55 | 74.85 | 5.79 | 50.28 |
| | CHED w/ KgCVAE | 78.62 | 77.21 | 4.77 | 48.80 |
| | CHED w/ AC | 83.66 | 80.41 | 4.80 | 47.77 |
| | CHED w/ RL | 87.94 | 85.79 | 4.87 | 50.32 |
| | CHED w/ IMPLICIT | 79.49 | 76.02 | 7.56 | 49.71 |
| | CHED w/ EXPLICIT | 85.50 | 83.25 | 7.26 | 49.42 |
| | CHED w/ ENSEMBLE | 86.17 | 82.86 | 7.41 | 49.71 |

Table 3.5: Controllability for each dialogue act (DA) label on DailyDialog (CHED w/ ENSEMBLE vs. CHED w/ SCE)

| DA | Prec. | Rec. | F1 (improv.) | Freq. |
|---|---|---|---|---|
| Inform | 89.62 | 93.06 | 91.31 (+**3.75**) | 3257 |
| Questions | 98.89 | 99.24 | 99.07 (+**1.18**) | 1713 |
| Directives | 94.71 | 85.17 | 89.69 (+**8.84**) | 1052 |
| Commissive | 68.40 | 66.02 | 67.19 (+**7.96**) | 718 |

Table 3.6: Controllability for each dialogue act (DA) label on SWBD (CHED w/ ENSEMBLE vs. CHED w/ SCE). This table shows only dialogue act labels with a frequency of 5 or more.

| DA | Prec. | Rec. | F1 (improv.) | Freq. |
|---|---|---|---|---|
| statement non opinion | 95.14 | 99.22 | 97.13 (+**0.97**) | 2169 |
| acknowledgements (backchannel) | 97.17 | 98.88 | 98.02 (+**0.96**) | 1250 |
| statement opinion | 97.88 | 86.94 | 92.08 (+**8.45**) | 689 |
| abandoned or turned exit/uninterpretable | 99.41 | 100.00 | 99.70 (+**0.44**) | 674 |
| yes/no question | 93.96 | 97.90 | 95.89 (+**18.76**) | 143 |
| appreciation | 96.40 | 100.00 | 98.17 (−0.34) | 134 |
| agree/accept | 84.69 | 80.58 | 82.59 (+**71.24**) | 103 |
| yes answers | 89.47 | 65.38 | 75.56 (+**3.65**) | 52 |
| wh question | 78.46 | 100.00 | 87.93 (−8.14) | 51 |
| backchannel in question form | 100.00 | 100.00 | 100.00 (+**1.15**) | 43 |
| conventional closing | 97.56 | 100.00 | 98.77 (+**3.65**) | 40 |
| response acknowledgement | 94.74 | 100.00 | 97.30 (+**37.30**) | 18 |
| open question | 66.67 | 100.00 | 80.00 (+**5.00**) | 18 |
| no answers | 78.57 | 73.33 | 75.86 (+**16.40**) | 15 |
| declarative yes/no question | 100.00 | 10.00 | 18.18 (+0.00) | 10 |
| affirmative non-yes answers | 50.00 | 75.00 | 60.00 (+**22.15**) | 8 |
| rhetorical questions | 0.00 | 0.00 | 0.00 (+0.00) | 8 |
| summarize or reformulate | 100.00 | 25.00 | 40.00 (+**40.00**) | 8 |
| action directive | 0.00 | 0.00 | 0.00 (−60.00) | 7 |
| other | 100.00 | 33.33 | 50.00 (+**5.56**) | 6 |

Table 3.7: Response quality of each model with different objective functions in human evaluations. This table shows percentages for each score. Avg is weighted by percentage of scores.

| Models | +2 | +1 | +0 | Avg |
|---|---|---|---|---|
| HED w/ SCE | 23.33 | 38.33 | 38.33 | 0.85 |
| CHED w/ SCE | 25.00 | 38.75 | 36.25 | 0.88 |
| CHED w/ RL | 24.58 | 36.66 | 38.75 | 0.85 |
| CHED w/ IMPLICIT | 27.91 | 43.33 | 28.75 | **0.99** |
| CHED w/ EXPLICIT | 25.41 | 39.58 | 35.00 | 0.90 |
| CHED w/ ENSEMBLE | 27.91 | 41.25 | 30.83 | 0.97 |

Table 3.8: Controllability of response generation: We calculated agreement (Cohen's $\kappa$) with automatic annotation results by dialogue act classifier. Table shows the filtered results of contradicted responses. Ratio is the proportion of the acceptable response that was properly controlled by dialogue acts.

| Models | Accuracy | Agreement | Ratio |
|---|---|---|---|
| CHED w/ SCE | 85.14 | 0.68 | 54.27 |
| CHED w/ RL | 91.54 | 0.82 | 56.06 |
| CHED w/ IMPLICIT | 87.84 | 0.72 | 62.58 |
| CHED w/ EXPLICIT | **93.06** | 0.69 | 60.48 |
| CHED w/ ENSEMBLE | 92.68 | 0.81 | **64.12** |

Table 3.9: Generated responses from models given a dialogue history and a dialogue act. A and B denote system and user. Reference utterances for dialogue context sampled from test set shown after the context. Generation results in contexts from models are shown after references. Before generation results, given dialogue acts are indicated by round brackets.

| | |
|---|---|
| **A** | (Commissive) yes of course . this is a really good one actually . |
| **B** | (Questions) how much does it cost ? |
| **A: Human** | (Inform) this one is 1500 yuan . |
| **A: HED** | (Inform) want to take a look at it . |
| **A: CHED** | (Inform) $ 200 , please . |
| **A: CHED w/ RL** | (Inform) $ 800 per month . |
| **A: CHED w/ IMPLICIT** | (Inform) $ 200 , please . |
| **A: CHED w/ EXPLICIT** | (Inform) $ 200 . |
| **A: CHED w/ ENSEMBLE** | (Inform) $ 200 , please . |
| **A** | (Questions) what ' s the problem , nada ? you look down in the dumps . |
| **B** | (Inform) i don ' t know . my life is a big mess . everything is so complicated . |
| **A: Human** | (Questions) come on , nothing can be that bad ? |
| **A: HED** | (Questions) what do you think is the most important thing for you ? |
| **A: CHED** | (Questions) what do you mean ? |
| **A: CHED w/ RL** | (Questions) what do you think of the most important thing in the city ? |
| **A: CHED w/ IMPLICIT** | (Questions) what do you mean ? |
| **A: CHED w/ EXPLICIT** | (Questions) what do you mean ? |
| **A: CHED w/ ENSEMBLE** | (Questions) what do you think is the most important thing for you to do ? |
| **A** | (Inform) it ' s getting late . i really have to go . |
| **B** | (Directives) i hope we ' ll get together again soon . |
| **A: Human** | (Commissive) i hope so . a good journey to you ! |
| **A: HED** | (Commissive) i don ' t know what to do . |
| **A: CHED** | (Commissive) i don ' t think so . |
| **A: CHED w/ RL** | (Commissive) ok . let ' s go . |
| **A: CHED w/ IMPLICIT** | (Commissive) ok . let ' s go . |
| **A: CHED w/ EXPLICIT** | (Commissive) i will . thank you for your help . |
| **A: CHED w/ ENSEMBLE** | (Commissive) i think so |
| **A** | (Directives) i ' d like to make an appointment with dr . cooper . could you arrange it for me ? |
| **B: Human** | (Directives) yes . would tomorrow morning be all right with you ? |
| **B: HED** | (Directives) what about this one ? |
| **B: CHED** | (Directives) i need to make a reservation for me , please . |
| **B: CHED w/ RL** | (Directives) i need to check out this form . can you give me a call , please ? |
| **B: CHED w/ IMPLICIT** | (Directives) sure . would you please fill out this form , please ? |
| **B: CHED w/ EXPLICIT** | (Directives) well , i would like to buy a ticket . |
| **B: CHED w/ ENSEMBLE** | (Directives) you need to check out this form . would you like to go with me ? |

# 4 Entrainable neural conversation model

In this chapter, we describe how to incorporate entrainment, an attractive human phenomenon, into neural conversation models.

## 4.1 Introduction

Entrainment is a well-known conversational phenomenon in which dialogue participants mutually synchronize with regards to various aspects: lexical choice [Brennan and Clark, 1996], syntax [Reitter and Moore, 2007], style [Niederhoffer and Pennebaker, 2002], acoustic prosody [Natale, 1975, Ward and Litman, 2007], turn-taking [Campbell and Scherer, 2010, Beňuš et al., 2014], and dialogue acts [Mizukami et al., 2016]. Entrainment has a high correlation with dialogue success, naturalness, and engagement [Nenkova et al., 2008, Levitan et al., 2015, Nasir et al., 2019]. Some existing works evaluated the dialogue quality and the performance of dialogue systems through entrainment analysis [Weiss, 2020]. Although phenomena related to entrainment suggest that the quality of human-human and human-machine dialogues can be improved, it remains challenging to build a dialogue system that can explicitly consider the entrainment phenomena in the framework of a neural conversation model, which has been actively studied in recent years [Vinyals and Le, 2015, Serban et al., 2016].

In this study, we incorporate entrainment phenomena into a neural conversation model for building a more natural and user-satisfied dialogue system. We construct a neural conversation model that can control the degree of entrainment as intention of generated responses based on a framework of reinforcement learning (RL) [Williams, 1992]. We define the automatic entrainment scores based on the local interpersonal distance [Nasir et al., 2019], which focuses on lexical entrainment. We use this score to optimize a neural conversation model by RL.

In Section 4.3, we describe our task of entrainable conversation modeling (Section 4.3.1), a conditional generation model based on conventional architecture (Section 4.3.2), and our proposed model optimized to entrainment scores by RL (Section 4.3.3). In experiments, we performed a preliminary analysis using the defined entrainment scores to clarify the relationship between user assessment and entrainment phenomena in a chit-chat dialogue domain (Section 4.4). Experimental results showed that our entrainment scores correlated with human assessment in human-human and human-machine dialogues in the chit-chat domain (Section 4.5). As a model evaluation, we conducted subjective and objective evaluations (Section 4.6). Our proposed model generated comparable or more natural responses compared with general neural conversation models, which optimized by word prediction based on cross-entropy loss, and controlled well the degree of entrainment of the generated responses (Section 4.7). We discuss the challenges for the advancement of entrained response generation in neural conversation models by analyzing our experimental results (Section 4.8).

## 4.2 Related work

Many studies have analyzed entrainment in dialogues and shown that we can observe the phenomena in dialogues from various aspects: lexical choice [Brennan and Clark, 1996], syntax [Reitter and Moore, 2007], style [Niederhoffer and Pennebaker, 2002], acoustic prosody [Natale, 1975], turn-taking [Campbell and Scherer, 2010], and dialogue acts [Mizukami et al., 2016]. Furthermore, automatic entrainment scores have been proposed that focuses on these aspects. These entrainment scores were highly correlated with dialogue success, naturalness, and engagement [Nenkova et al., 2008, Levitan et al., 2015, Nasir et al., 2019].

Some studies used the knowledge obtained by analyzing the entrainment to the dialogue system. One work [Campbell and Scherer, 2010] predicted the user's turn-taking behavior by considering entrainment. Another work [Fandrianto and Eskenazi, 2012] modeled a dialogue strategy to intentionally increase the accuracy of the automatic speech recognition using entrainment, and another [Levitan, 2013] unified these works. Although these studies were conducted on modeling and predicting the entrainment of the user's behaviors, it remains challenging problem to build a dialogue system that can make entrainment to users to improve the dialogue system's response quality. In other words, insufficient studies have positively affected users through entrainment by the system.

On the other hand, recent neural conversation models focus on the efficient use and encoding of dialogue history [Serban et al., 2016, Tian et al., 2017]. However, they do not directly handle entrainment phenomena because they are achieved by minimizing the cross-entropy loss of word prediction in decoder networks. Both the model networks that consider the dialogue context and the objective function of the model itself must be improved to achieve entrainable response generation.

In this study, we introduce a reinforcement learning (RL) framework [Williams, 1992, Ranzato et al., 2015] to optimize a neural conversation model for automatic entrainment scores. Entrainment scores are given as RL rewards that enable neural conversation models to generate appropriately entrained responses for their dialogue contexts. Existing studies have already described the correlation between human assessments and automatic entrainment scores. We performed a follow-up analysis using chit-chat dialogue corpora to confirm that we can use the scores as an objective function. By optimizing the model to maximizing these scores, we expect that our neural conversation model can generate more natural responses.

## 4.3 Entrainable neural conversation model

In this study, we focus on lexical entrainment, which is related to lexical choice in dialogues. We are particularly interested in semantic entrainment, which is a variant of lexical entrainment, that considers the similarity of words in semantic space, not only the surface agreement of selected words. We introduce an entrainment score based on the similarity in semantic spaces in word-distributed representation [Kusner et al., 2015, Nasir et al., 2019] to capture local entrainment trends for each turn in the dialogue. We optimize the neural conversation model using RL to maximize the entrainment scores of the generated responses. We formulate the problem as conditional neural conversation modeling, which uses the degree of entrainment of the response as a condition, because generating a highly entrained response is not always appropriate even though the model has to control the degree of entrainment based on the dialogue contexts.

In this section, we first describe an overview of the response generation task tackled in this study (Section 4.3.1). Then we describe the architecture of a conditional neural conversation model given the degree of entrainment (Section 4.3.2). Finally, we describe a method that optimizes the conditional neural conversation model using RL to fit the given degree of entrainment (Section 4.3.3).

### 4.3.1 Task definition

We formally define the task of entrainable conversation modeling as a response generation task given a dialogue context and a degree of entrainment to the dialogue context. Define generated response word sequence $R = [w_1, w_2, \cdots, w_T]$, given dialogue context $H = [H_1, H_2, \ldots, H_N]$ and degree of entrainment of target response $r_{\text{target}} \in \mathbb{R}$. $N$ is the dialogue length, and $T$ is the number of words in an utterance.

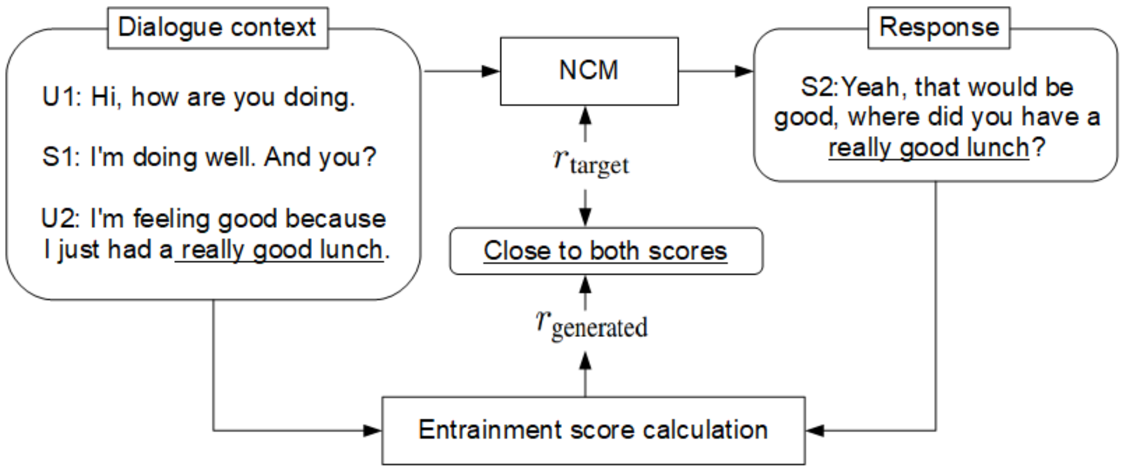



Figure 4.1: Task of entrainable conversation generation

In this setting, response $R$ is required to satisfy not only the appropriateness to the dialogue context but also the degree of entrainment to the dialogue context (Fig. 4.1). In other words, the neural conversation model enforces entrainment degree $r_{\text{generated}} \in \mathbb{R}$ of the actual generated response to be closer to indicated entrainment degree $r_{\text{target}}$. This optimization is achieved by minimizing the relative error of both entrainment degrees:

$$\underset{r_{\text{generated}} \in \mathbb{R}}{\text{minimize}} \ \text{relative\_error}(r_{\text{target}}, r_{\text{generated}}). \tag{4.1}$$

As an approach to building such neural conversation models controllable by a given condition, such as the entrainment degree, vector concatenation is widely used between a word vector and the vectorized condition to feed a decoder input [Li et al., 2016b, Huang et al., 2018a]. Some other works proposed to extend models [Peng et al., 2019, Zhou et al., 2019] for conditional generation according to given emotion labels in the task of the emotional dialogue generation.

### 4.3.2 Neural conversation model based on entrainment degree

We introduce a conditional neural conversation model based on a hierarchical encoder-decoder model [Serban et al., 2016] with a context attention mechanism, which explicitly gives an embedded vector of entrainment degree to the decoder (Fig. 4.2). We apply the vector concatenation as a widely used method for conditioning the decoder.

The encoder network has a hierarchical structure that consists of utterance and context encoders. The utterance encoder receives a word at each time step using forward RNNs to encode an utterance into a fixed-length vector:

$$h_{i,t} = \text{RNN}_{\text{utterance}}(h_{i,t-1}, \text{Embed}(w_{i,t})). \tag{4.2}$$

Here $i$ is the number of turns in the dialogue context, and $h_{i,t}$ is the hidden vector obtained by inputting each word $w_{i,t}$ in utterance $H_i$. Each word $w_{i,t}$, which is encoded to a fixed-length vector using an embedding layer, is used as input.

In the context encoder, utterance vectors are input to encode the dialogue history:

$$c_i = \text{RNN}_{\text{context}}(c_{i-1}, h_{i,T}). \tag{4.3}$$

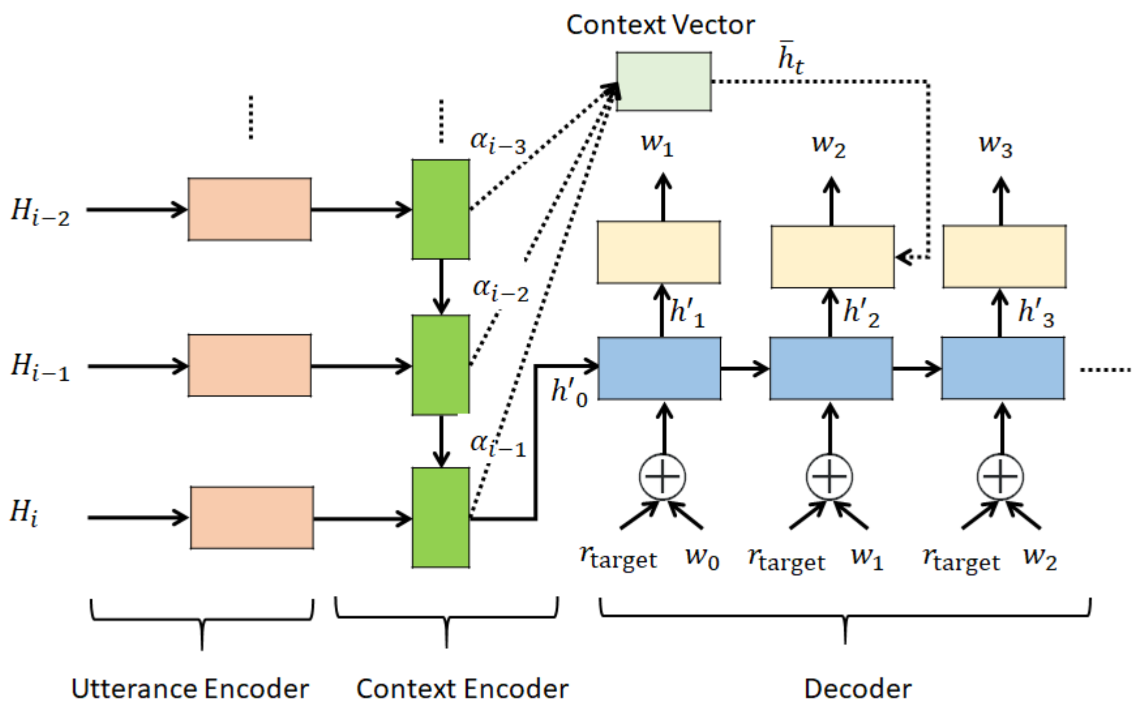



Figure 4.2: Neural conversation model with entrainment degree as a condition

Here $h_{i,T}$ is a hidden vector obtained at the last step in the encoding for each utterance. Resultant vector $c_N$ is fed into the decoder to generate a response sentence as initial hidden state $h'_0$. In the decoder, hidden state $h'_t$ of the decoder and the output probability of word $p_t$ are calculated:

$$h'_t = \mathrm{RNN}_{\mathrm{dec}}(h'_{t-1}, [\mathrm{Embed}(w_{t-1}); \mathrm{Linear}_{\mathrm{ent}}(r_{\mathrm{target}})]), \tag{4.4}$$

$$p_t = \mathrm{softmax}(\mathrm{Linear}_{\mathrm{proj}}(h'_t)). \tag{4.5}$$

Here $\mathrm{Linear}_{\mathrm{proj}}$ is a projection layer, which maps $h'_t$ to a vector of vocabulary size $|\mathcal{V}|$. $\mathrm{Linear}_{\mathrm{ent}}$ is a linear transformation layer that embeds target entrainment degree $r_{\mathrm{target}}$ into a fixed-length vector. $w_{i,t}$ is sampled from $p_t$ and used as a part of the input for the next step. In this decoding architecture, we expect the decoder to generate a response with an appropriate degree of entrainment for the dialogue history by also inputting entrainment degree $r_{\mathrm{target}}$ in addition to already generated words. Note that we used teacher-forcing in the training process [Vinyals and Le, 2015].

We also introduce a simple attention mechanism to the above decoder for efficiently handling the information from the encoded dialogue context. Specifically, let $c_{1:N}$ be a sequence of vectors obtained by the context encoder, and let $h'_t$ be the hidden states of the decoder in the $t$ step. We compute the alignment weights based on general-attention [Luong et al., 2015] for each hidden state and obtain context vector $\bar{h}_t$:

$$\alpha_j = \frac{\exp(\mathrm{score}(c_j, h'_t))}{\sum_{\tilde{j}=1}^{N} \exp(\mathrm{score}(c_{\tilde{j}}, h'_t)}, \tag{4.6}$$

$$\bar{h}_t = \sum_{\tilde{j}=1}^{N} \alpha_{\tilde{j}} \cdot c_{\tilde{j}}. \tag{4.7}$$

The output words in step $t$ are predicted using computed context vector $\bar{h}_t$:

$$\hat{h}_t = \tanh(\mathrm{Linear}_{\mathrm{attn}}([\bar{h}_t; h'_t])), \tag{4.8}$$

$$p_t = \mathrm{softmax}(\mathrm{Linear}_{\mathrm{proj}}(\hat{h}_t)). \tag{4.9}$$

In general, training the neural conversation model is based on minimizing the cross-entropy:

$$L_{\mathrm{CE}} = -\sum_{t=1}^{T} \log \frac{\exp(x_{t,e})}{\sum_{k}^{|\mathcal{V}|} \exp(x_{t,k})}. \tag{4.10}$$

Here $|\mathcal{V}|$ denotes the vocabulary size, $x_t \in \mathbb{R}^{|\mathcal{V}|}$ denotes the output of the projection layer in the decoding steps, and $x_{t,e} \in \mathbb{R}^{|\mathcal{V}|}$ denotes the $e$-th element that correspond to target word $w_t$.

However, perhaps models based on minimizing cross-entropy loss do not efficiently use the information in the dialogue context [Sankar et al., 2019]. Furthermore, since cross-entropy loss is not designed to handle entrainment phenomena, we have to define a new objective function for building a neural conversation model that is optimized to entrainment scores.

### 4.3.3 Model optimization to entrainment degree based on reinforcement learning

Our final goal is to build an entrainable neural conversational model based on the given entrainment degree. However, model optimization based on existing cross-entropy loss does not satisfactorily control the generation because the optimization is calculated word-by-word. In contrast, optimization based on reinforcement learning has the potential to train such a controllable response generation model [Kawano et al., 2019]. Thus, we introduce the REINFORCE algorithm, which is based on reinforcement learning [Williams, 1992, Ranzato et al., 2015].

In this section, we describe the objective function and its optimization method using the REINFORCE algorithm (Section 4.3.3). We introduce a reward function using an automatic entrainment score to optimize the model (Section 4.3.3) and scrutinize the training procedure for our neural conversation model based on reinforcement learning (Section 4.3.3).

#### REINFORCE algorithm

The generation process in the neural conversation model is formalized as a Markov decision process (MDP) and optimized with reinforcement learning (RL) [Li et al., 2017a]. The problem of response generation in the neural conversation model is generating response word sequence $R = [w_1, w_2, \cdots, w_T]$ that corresponds to given dialogue history $H$ and target entrainment degree $r_{\text{target}}$. Formally the generation process is defined as choosing an action to generate word $w_t$ given a state, already generated words $[w_1, w_2, \cdots, w_{t-1}]$, in time-step $t$ [Yu et al., 2017]. Such a word selection process in the generation is defined as an action sequence, which is generated by an actual policy in MDP.

We define a reward function based on entrainment scores to encourage the model to generate entrained responses. The entrainment's evaluation score is fed as a reward to update the generator's policy in the RL. We use a policy gradient (REINFORCE algorithm) [Williams, 1992, Ranzato et al., 2015] to train the policy. The gradient of objective function is defined:

$$
\begin{aligned}
\nabla J_{\mathrm{RL}}(\theta) \simeq & \frac{1}{T}\sum_{t=1}^{T}\sum_{w_t\in\mathcal{V}} Q^{G_\theta}(w_{1:t-1}, w_t) \\
& \cdot \nabla_\theta G_\theta(w_t|w_{1:t-1}) \qquad (4.11) \\
= & \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{w_t\sim G_\theta}[Q^{G_\theta}(w_{1:t-1}, w_t) \\
& \cdot \nabla_\theta \log p(w_t|w_{1:t-1})]. \qquad (4.12)
\end{aligned}
$$

Here $\theta$ is a parameter of the policy. $\mathcal{V}$ is a vocabulary, $w_{1:t-1}$ indicates the already generated word sequence (state in MDP), and $p(w_t|w_{1:t-1}) = G_\theta(w_t|w_{1:t-1})$ is the generative probability of word $w_t \in \mathcal{V}$ (action in MDP) in the decoder. $Q^{G_\theta}(w_{1:t-1}, w_t)$ is an action-value function that gives an expected future reward when the system generates word $w_t$ given the state: already generated word sequence $w_{1:t-1}$. The expectation $\mathbb{E}[\cdot]$ can be approximated by sampling.

The action-value function for each step is approximately calculated using a Monte Carlo tree search (MCTS) [Browne et al., 2012, Yu et al., 2017] under the current policy and its parameter $\theta$:

$$
\begin{aligned}
& Q^{G_\theta}(R_{1:t-1}, w_t) = \qquad (4.13) \\
& \begin{cases} \frac{1}{N}\sum_{n=1}^{N} \mathrm{r}(H, R^n_{1:t}, R^{\mathrm{ref}}, r_{\mathrm{target}}) & \text{for} \quad t < T, \\ \mathrm{r}(H, R_{1:t}, R^{\mathrm{ref}}, r_{\mathrm{target}}) & \text{for} \quad t = T. \end{cases}
\end{aligned}
$$

Here $\mathrm{r}(\cdot)$ is a reward function that evaluates the entrainment degree of response $R_{1:T} = \{w_1, w_2, \cdots, w_T\}$. $R^n_{1:t}$ is the generated response using a roll-out [Yu et al., 2017] from partial-generated response $R_{1:t}$ using parameter $\theta$. $R^{\mathrm{ref}}$ is a reference response, and $n$ is the number of roll-outs*. This reward function calculates the reward based on the relative error between a given entrainment degree and the entrainment degree of the actual generated response to allow it to control the entrainment degree of the generated response. Note that we can use an arbitrary score in this formulation. We use a reliable entrainment score based on the similarity of the semantic space of the words to feed the entrainment degree (score) as the reward.

*We set the number of roll-outs to 5. However, since the computation cost of MCTS is high when training the large model, we can also adopt an approach for speeding up the training, such as REGS [Li et al., 2017a], instead of MCTS.

### Reward function for evaluating entrainment degree

We construct a reward function based on the idea of a local interpersonal distance (LID), which is a previously proposed turn-level entrainment score [Nasir et al., 2019]. LID uses a predefined number of turns (context lengths) in response to the utterances of the primary speaker (anchor). The anchor utterance and response pair that has a minimum distance is chosen to calculate LID. This calculation is based on local entrainment, which is not always observed in the immediate response to the primary speaker's turn. It might be sustained and exhibited after a few turns [Pickering and Garrod, 2006]. In this study, unlike the LID's original definition [Nasir et al., 2019], we calculate the similarity between the anchor utterance and each past contextual utterance by another speaker and choose an anchor and contextual utterance pair with minimum distance. However, note that there is no difference in the nature of both scores.

To calculate the distance between two utterances, we use Word Mover's Distance (WMD) [Kusner et al., 2015], which is calculated from the distributed vector representations of words in a document. WMD targets both semantic and syntactic information to get a distance between text documents. WMD calculates the Earth Mover's Distance [Rubner et al., 1998] between sets of word vectors that are contained in the target sentences (documents). This calculation is based on the minimum travel distance. Specifically, let $e_i \in \mathbb{R}^d$ represents $i$-th word, as defined by word-embedding $E \in \mathbb{R}^{d \times n}$ for vocabulary of $n$ words. We also define $a$ and $b$ are $n$-dimensional normalized vectors, which consist of bag-of-words of two sentences. $a_i$ indicates the count of the word i in the sentence[†]. The WMD introduces an transport matrix $T \in \mathbb{R}^{n \times n}$, such that $T_{i,j}$ indicates how much of $a_i$ should be transported to $b_j$. Formally, the WMD learns $T$ to minimize:

$$\begin{aligned} \mathrm{WMD}(a,b) = \min_{T \geq 0} & \sum_{i=1}^{n}\sum_{j=1}^{n} T_{ij}||e_i - e_j|| \\ \text{subject to} \quad & \sum_{j=1}^{n} T_{ij} = a_i \quad \forall i, \\ \text{and} \quad & \sum_{i=1}^{n} T_{ij} = b_j \quad \forall j. \end{aligned} \tag{4.14}$$

[†]Note that $a_i$ is normalized over all words in $a$.

To solve this minimization problem, we used the efficient implementations [Pele and Werman, 2008,Pele and M.Werman, 2009,Kusner et al., 2015][‡]. In this study, we calculate WMD using all words within the utterance because we intended to consider the words related to speaking style, not only content words within utterance in the similarity calculation. The similarity evaluation is closed within the two utterances, and the properties of the words are only considered in a word embedding. However, according to various entrainment purposes, we may need to evaluate the word similarity considering various aspects, such as the importance of co-occurrence of proper nouns or rare words. In such a case, our method can replace WMD with another specially designed similarity function consider word importance.

We define target entrainment degree $r_{\text{target}}$ based on the idea of LID:

$$\text{sim}(x, y) = \mathrm{e}^{-\text{WMD}(\text{bow}(x), \text{bow}(y))^2}, \tag{4.15}$$

$$r_{\text{target}} = \mathrm{r}_{\text{target}}(H, R^{\text{ref}}) = \max_{H_j^{\text{other}} \in H^{\text{other}} \subset H} \text{sim}(H_j^{\text{other}}, R^{\text{ref}}). \tag{4.16}$$

Here the sim$(\cdot)$ function normalizes the calculation results of WMD as the similarities between utterances ($x$ and $y$). $e$ is a natural logarithm, and bow$(\cdot)$ is a function to convert the given sentence to bag-of-words representation. $R^{\text{ref}}$ is the reference response corresponding to dialogue history $H$, and $H^{\text{other}} \subset H$ is a set consisting of the most recent $k$ utterances from the dialogue history $H$, excluding the primary speaker's utterances. $r_{\text{target}}$ is assumed to be a similarity that takes values from 0 to 1.

Next we define entrainment degree $r_{\text{generated}}$ of the actual generated response:

$$u_{\text{target}} = \mathrm{u}_{\text{target}}(H, R^{\text{ref}}) = \operatorname*{arg\,max}_{H_j^{\text{other}} \in H^{\text{other}} \subset H} \text{sim}(H_j^{\text{other}}, R^{\text{ref}}), \tag{4.17}$$

$$r_{\text{generated}} = \mathrm{r}_{\text{generated}}(H, R^{\text{ref}}, R) = \text{sim}(R, u_{\text{target}}). \tag{4.18}$$

Here $u_{\text{target}}$ is a target utterance to make entrainment by system generation, which has the maximum similarity of every pair formed by the anchor utterance of the

[‡]https://github.com/RaRe-Technologies/gensim

reference and a context utterance. Thus, $r_{\text{generated}}$ is calculated as the similarity between the generated response and the target.

The reward given to the generated response is calculated from the relative error between $r_{\text{target}}$ and $r_{\text{generated}}$:

$$r = \mathrm{r}(H, R, R^{\text{ref}}, r_{\text{target}}) = 1 - \frac{|r_{\text{target}} - r_{\text{generated}}|}{\max(r_{\text{target}}, 1 - r_{\text{target}})}. \tag{4.19}$$

This reward function gives more reward when the relative error between the entrainment degree of the generated response and the indicated entrainment degree is small. In other words, it gives penalty if the generated utterance is over or under-entrained compared with the reference.

We used different functions for $r_{\text{taget}}$ and $r_{\text{generated}}$ because using the same function will lead to learning a lazy policy that always refers to the previous utterance.

### Model training based on REINFORCE algorithm

The training procedure of a neural conversation model with RL is shown in Algorithm 2.

**Algorithm 2** Training Procedure

**Require:** generator policy $G_\theta$; roll-out policy $G'_\theta$

Initialize $G_\theta$ with random weights $\theta$<br>
Pretrain $G_\theta$ to minimize $L_{\text{CE}}$ ▷ (4.10)<br>
**for** number of iterations **do**<br>
    $G'_\theta \leftarrow G_\theta$<br>
    **for** number of steps **do**<br>
        sample $(H, R^{\text{ref}}, r_{\text{target}})$ from training data<br>
        generate response $R$ using $G'_\theta$ on $H$ and $r_{\text{target}}$<br>
        compute $Q^{G_\theta}$ for $(H, R, r_{\text{target}})$ using $G'_\theta$ ▷ (4.13)<br>
        update $G_\theta$ based on $J_{\text{RL}}(\theta)$ ▷ (4.12)<br>
    **end for**<br>
**end for**

First, we pre-train the neural conversation model by minimizing cross-entropy loss $L_{\text{CE}}$. Then we train it to maximize objective function $J_{\text{RL}}(\theta)$ using reinforcement learning and add $L_{\text{CE}}$ to the loss to stabilize the training. This approach works as a teacher forcing and prevents the collapse of policies due to the model's inability to access the reference response [Li et al., 2017a]. The policy used to calculate $Q^{G_\theta}$ is updated every 20 steps. We use the model with the highest reward sum for the generated response and the deterioration of perplexity within 1.0 points in the validation set for the evaluation.

## 4.4 Entrainment analysis setting

LID, which is used as a reward in this study, probably correlates with human assessment (therapeutic outcomes and affective behaviors) in the dialogues of clinical psychology and psychotherapy [Nasir et al., 2019]. On the other hand, no examination has focused on chit-chat dialogues, which are the main focus of this study. Therefore, we performed a preliminary analysis to clarify the relationship between user assessment and entrainment in chit-chat dialogues.

We used Spearman's rank correlation coefficient to evaluate the relationship between the Conversational Linguistic Distance (CLiD) [Nasir et al., 2019], a dialogue session-level entrainment score calculated by the mean of the LIDs, and the user assessment assigned to each dialogue. CLiD is defined:

$$\mathrm{CLiD}(\mathcal{D}) = \frac{\sum_{(H,R^{\mathrm{ref}})\in\mathcal{D}} \mathrm{r}_{\mathrm{target}}(H, R^{\mathrm{ref}})}{|\mathcal{D}|}. \tag{4.20}$$

We applied (4.16) to each turn of the dialogues and averaged the results as the dialogue level entrainment scores. Here, $(H, R^{\mathrm{ref}}) \in \mathcal{D}$ is a context and response pair for each turn in the dialogue. Note that the definition changes from the original CLiD to fit our problem.

For the entrainment analysis, we used the following chit-chat dialogue corpora (Table 4.1):

- **ConvAI2-wild-evaluation**: Dialogues between a human and a system that participated in the ConvAI2 competition[§]. Each dialogue was evaluated by human participants on a five-point scale.
- **NTT-chit-chat** : Dialogues between human participants that covered as

[§]http://convai.io/data

wide range of topics [Arimoto et al., 2019]. Participants in each dialogue evaluated it from the following three viewpoints on a seven-point Likert scale: 1) strongly disagree; 2) disagree; 3) slightly disagree; 4) neither; 5) slightly agree; 6) agree; 7) strongly agree.

- Q1: “I am satisfied with the current dialogue. I’d like to have such a dialogue again.”
- Q2: “I found myself interested in the topic of the current dialogue.”
- Q3: “In the current dialogue, I spoke positively on my own.”

Table 4.1: Number of dialogues/utterances in each corpus

| Corpus | Dialogues | Utterances |
|---|---|---|
| ConvAI2-wild-evaluation | 2,483 | 41,415 |
| NTT-chit-chat | 3,483 | 56,566 |

## 4.5 Entrainment analysis results

We performed a correlation analysis of ConvAI2-wild-evaluation, as shown in Table 4.2. Here $\rho$ is the correlation calculated by Spearman’s rank correlation analysis, and the $p$-value is the probability of the null hypothesis. We calculated the CLiD for two types based on their attributes because the speaker has distinctly different attributes: “Human $\rightarrow$ System” shows the human responses to the system, and “System $\rightarrow$ Human” shows the system responses to the humans.

Table 4.2: Entrainment analysis results for ConvAI2-wild-evaluation

| Types | $k=1$ | | $k=2$ | | $k=3$ | |
|---|---|---|---|---|---|---|
| | $\rho$ | $p$-value | $\rho$ | $p$-value | $\rho$ | $p$-value |
| Human $\rightarrow$ System | 0.19 | $6.25 \times 10^{-22}$ | 0.24 | $2.83 \times 10^{-33}$ | 0.26 | $3.03 \times 10^{-41}$ |
| System $\rightarrow$ Human | 0.22 | $9.22 \times 10^{-31}$ | 0.23 | $5.51 \times 10^{-31}$ | 0.23 | $2.90 \times 10^{-33}$ |

As shown in Table 4.2, we confirmed that CLiD has a positive correlation with human scores, regardless of any setting used to calculate it. This result indicates that the entrainment degree is critical for improving the quality of human-machine dialogues. Since many systems based on a neural network are not able to handle the dialogue context [Sankar et al., 2019], this result might be deeply related not only to entrainment but also to whether the system can generate a context-relevant response. To compare cases using different values of $k$, we confirmed a stronger correlation in the case of $k = 2$ and $k = 3$. In fact, humans often respond with an awareness of both the previous utterances but also deeper utterances from the past in a dialogue history [Pickering and Garrod, 2006].

We also performed a correlation analysis between average assessments by two participants and CLiD in an NTT-chit-chat (Table 4.3).

Table 4.3: Entrainment analysis results for NTT-chit-chat

| Question | $k = 1$ | | $k = 2$ | |
|---|---|---|---|---|
| | $\rho$ | $p$-value | $\rho$ | $p$-value |
| Q1 | 0.13 | $2.45 \times 10^{-15}$ | 0.10 | $7.13 \times 10^{-10}$ |
| Q2 | 0.11 | $6.30 \times 10^{-12}$ | 0.08 | $4.11 \times 10^{-7}$ |
| Q3 | 0.09 | $4.19 \times 10^{-8}$ | 0.07 | $2.66 \times 10^{-5}$ |

Table 4.3 shows a moderately positive correlation between the user assessments corresponding to these questions and the CLiD. This result suggests that using entrainment in dialogues is an effective strategy to improve user satisfaction in chit-chat dialogues. As stronger correlation is observed in $k = 1$ than in $k = 2$. This is probably because NTT-chit-chat has multiple utterances per turn. In other words, when we apply CLiD to dialogues that contain a lot of information in one turn, it is difficult to find strong correlations between CLiD and human scores, because LIDs, which are CLiD components, will be biased by the number of words in the utterances.

These results indicate that CLiD, which is calculated by LID averages, is a useful and strict score to evaluate dialogues in the chit-chat domain, if their utterance length is limited. In other words, these results support our hypothesis: maximizing the LIDs in dialogues increases dialogue quality.

## 4.6 Evaluation setting for response generation

We confirmed that improving LID scores is important in human-machine dialogue setting as well, which is a basic hypothesis of our entrainable dialogue modeling. In this section, we describe the experimental settings to confirm the effect of our proposed entrainable neural conversation model.

### 4.6.1 Dataset

We used the dataset provided at the ConvAI2 competition, which was also used to train the system in the ConvAI2-wild-evaluation described in Section 4.4. This dataset was divided into train, validation, and test sets (Table 4.4). We divided the original development data into validation and test sets[¶].

Table 4.4: Number of dialogues/utterances in ConvAI2 dataset

| | Dialogues | Utterances |
|---|---|---|
| Train | 17,876 | 262,862 |
| Validation | 498 | 7,798 |
| Test | 499 | 7,788 |

### 4.6.2 Competing models

We compared the following different types of neural conversation models in our evaluations:

- **ASEQ2SEQ**: a standard neural conversation model that encodes a previous utterance as a query for decoding a response with an attention mechanism (general-attention) [Luong et al., 2015].
- **HED**: a hierarchical encoder-decoder model [Serban et al., 2016] without an attention mechanism and conditioning to a decoder.
- **AHED**: a model that combines an attention mechanism with the HED model.

[¶]Note that the original test set in ConvAI2 dataset are private.

- **C-ASEQ2SEQ**: a model with conditioning based on ASEQ2SEQ. We gave the condition (degree of entrainment) as described in Section 4.3.2.
- **C-HED**: a HED model with a conditioning mechanism. We gave the condition (degree of entrainment) as described in Section 4.3.2.
- **C-AHED**: an AHED model with conditioning. We gave the condition (degree of entrainment) as described in Section 4.3.2.

We trained these neural conversation models using conventional cross-entropy loss (**+CE**) and our proposed optimization based on reinforcement learning (**+RL**). We used the entrainment scores as a condition given to the decoder and explored the case with different $k \in \{1, 2, 3\}$ for score calculation. When $k = 1$, the entrainment score is calculated using only the previous utterance; when $k = 2$, it is calculated using the two most recent utterances by a non-primary speaker. Since the dialogue is performed alternately by two speakers, the neural conversation model needs to handle at most four utterances in a dialogue history when $k = 2$. However, the subjective evaluation is limited to $k = 1$ and $k = 2$ in this experiment because $k = 3$ did not show much difference compared with the other $k$ settings.

### 4.6.3 Hyper-parameter settings

We used the same hyper-parameter settings in these models. The vocabulary size was 15000, the word embedding size was 300, the entrainment embedding size was 50, the hidden-size was 500. We used a two-layer Gated Recurrent Unit (GRU) [Cho et al., 2014] as an RNN. In the training, we used a mini-batch size of 128, and an Adam optimizer [Kingma and Ba, 2014] with a learning rate of 1e-4. For the WMD calculation, we used pre-trained, word-distributed vectors[‖], which normalized the norm to 1 for each. We set the maximum length of the dialogue history to 4.

[‖] For English data: http://nlp.stanford.edu/projects/glove/
and for Japanese data: http://www.worksap.co.jp/nlp-activity/word-vector/

### 4.6.4 Automatic evaluations

We automatically evaluated the generation results using references in the test set. We used a beam search (a beam width of 5) for generating examples to be evaluated. For automatic evaluation, we used the following five different metrics:

- **Perplexity** (PPL) is a general metric for evaluating a language model performance. The model Likelihoods of the reference responses are calculated. Note that the perplexity scores do not directly reflect the quality of generation; for example, dull responses also have good perplexity scores.

- **BLEU**, which is the most popular automatic evaluation metric of language generation tasks, calculates the similarity between references and generated responses [Papineni et al., 2002] based on n-gram precision. We used BLEU2, which considers uni-grams and bi-grams because matches in higher-order n-grams are rarely observed response generation tasks.

- $\overline{\textbf{WMD}}$ is the average similarity between the references and the generated responses for each case in the test set. The similarity is calculated based on (4.15). The score is multiplied by 100 and displayed in a range of 0 to 100.

- $\bar{\mathbf{r}}$ is an average reward calculated from (4.19) to each generation. When this score is high, the entrainment degree of the generated response shows a similar degree to the reference. In other words, it shows that the neural conversation model controlled the response content well based on the entrainment degree. We sorted the test sets by the entertainment scores of the references and divided them into four parts to calculate $\bar{\mathrm{r}}$ for each ($\bar{\mathbf{r}}^{0\sim25\%}$, $\bar{\mathbf{r}}^{25\sim50\%}$, $\bar{\mathbf{r}}^{50\sim75\%}$, $\bar{\mathbf{r}}^{75\sim100\%}$). For example, $\bar{\mathrm{r}}^{0\sim25\%}$ shows the average reward of examples that have less entrainment scores in the references. These scores are multiplied by 100 and displayed in a range of 0 to 100.

- **Entropy** (Ent) is a diversity metric [Zhang et al., 2018] that reflects the evenness of the empirical n-gram distribution for the given responses: $Ent = \frac{1}{\sum_{w\in\mathcal{V}} C(w)} \sum_{w\in\mathcal{V}} C(w) \log \frac{C(w)}{\sum_{w'\in\mathcal{V}} C(w')}$, where, $\mathcal{V}$ is the set of all n-grams in the given responses, and $C(w)$ denotes the frequency of n-gram $w$. We set the uni-gram for evaluation.

### 4.6.5 Human subjective evaluations

Automatic evaluation scores still have a problem since they do not have high correlation with human subjective evaluation results [Liu et al., 2016]. Thus, we also examined models with a human subjective evaluation to confirm the naturalness of the generated responses. In the evaluation of naturalness, we used a 3-point scale in accordance with an existing work [Xing et al., 2017]. 240 generated responses were randomly selected from the test set, and human annotators added their evaluation scores for each sample by looking at the dialogue contexts. Detailed descriptions follow.

- **+2**: This response is not only relevant and natural, but also informative and interesting.
- **+1**: This response can be used as a response to the context, although it is universal, like "Yes, I see," "Me too," or "I don't know."
- **0**: This response cannot be used as a response to this context. It is either semantically irrelevant or dis-fluent.

Three annotators evaluated each sample, and the final score was decided by a simple majority. If the evaluation was completely disagreed (0, +1, and +2), the example was scored as 1.

Table 4.5: Automatic evaluation results for each neural conversation model

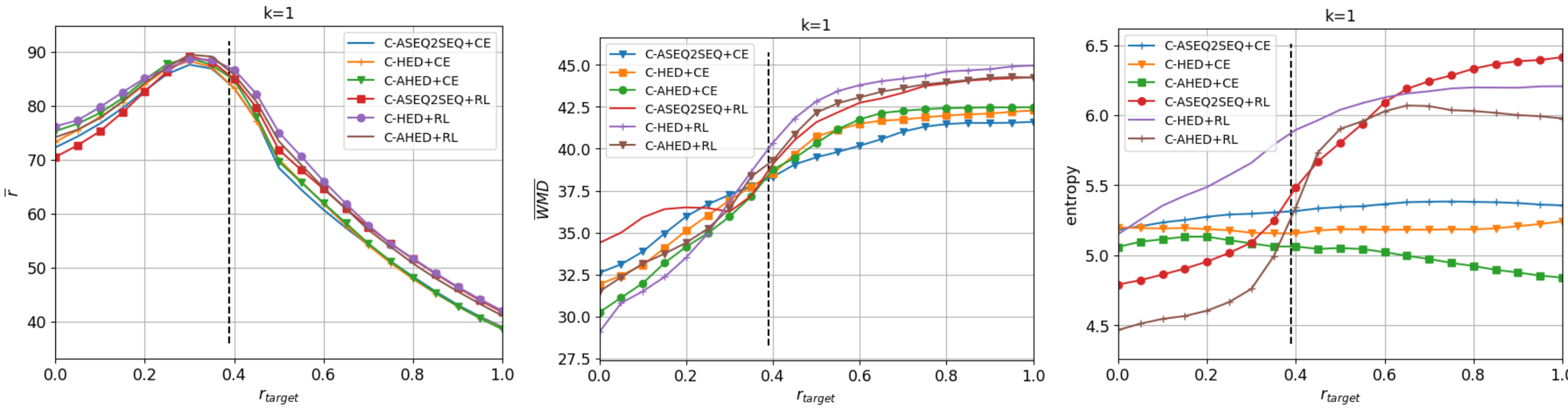
| Models ($k = 1$) | PPL | BLEU2 | $\overline{\text{WMD}}$ | $\overline{\text{r}}^{0\sim25\%}$ | $\overline{\text{r}}^{25\sim50\%}$ | $\overline{\text{r}}^{50\sim75\%}$ | $\overline{\text{r}}^{75\sim100\%}$ | $\overline{\text{r}}$ | Ent |
|---|---|---|---|---|---|---|---|---|---|
| ASEQ2SEQ+CE | 40.80 | **7.82** | 38.77 | 91.22 | 88.24 | 82.80 | 62.35 | 81.15 | 5.03 |
| HED+CE | 37.53 | 6.62 | 36.40 | 90.72 | 89.21 | 83.79 | 63.06 | 81.69 | 5.21 |
| AHED+CE | 39.11 | 6.71 | 38.16 | 91.04 | 88.44 | 82.40 | 62.11 | 81.00 | 5.11 |
| C-ASEQ2SEQ+CE | 40.05 | 3.69 | 38.77 | 90.90 | 88.08 | 84.19 | 68.82 | 83.01 | 5.33 |
| C-HED+CE | 37.12 | 6.56 | 39.17 | 91.41 | 88.52 | 84.25 | 70.04 | 83.55 | 5.20 |
| C-AHED+CE | 39.91 | 6.64 | 38.84 | 91.58 | 89.25 | 85.00 | 70.02 | 83.96 | 5.11 |
| C-ASEQ2SEQ+RL | 41.41 | 7.28 | 39.59 | 91.60 | 89.36 | 86.12 | 72.65 | 84.93 | 5.59 |
| C-HED+RL | 37.54 | 6.69 | **40.57** | 91.72 | 90.51 | **88.01** | **74.53** | **86.19** | **5.97** |
| C-AHED+RL | 40.01 | 7.42 | 39.96 | **91.84** | **90.53** | 86.36 | 72.75 | 85.37 | 5.52 |
| Models ($k = 2$) | PPL | BLEU2 | $\overline{\text{WMD}}$ | $\overline{\text{r}}^{0\sim25\%}$ | $\overline{\text{r}}^{25\sim50\%}$ | $\overline{\text{r}}^{50\sim75\%}$ | $\overline{\text{r}}^{75\sim100\%}$ | $\overline{\text{r}}$ | Ent |
| HED+CE | 37.53 | 6.62 | 36.40 | 87.99 | 86.69 | 82.45 | 62.74 | 79.97 | 5.21 |
| AHED+CE | 39.11 | 6.71 | 38.16 | 88.11 | 86.21 | 81.14 | 61.86 | 79.33 | 5.11 |
| C-HED+CE | 37.01 | 7.28 | 39.89 | 90.79 | 88.84 | 84.77 | 70.15 | 83.64 | 5.17 |
| C-AHED+CE | 38.76 | 5.17 | 38.43 | 89.49 | 87.64 | 84.44 | 67.54 | 82.27 | 5.18 |
| C-HED+RL | 37.24 | 6.39 | **40.67** | 90.25 | 89.49 | 86.91 | **74.18** | 85.21 | 5.62 |
| C-AHED+RL | 40.79 | **7.30** | 40.48 | **90.93** | **91.24** | **87.14** | 72.63 | **85.49** | **5.87** |
| Models ($k = 3$) | PPL | BLEU2 | $\overline{\text{WMD}}$ | $\overline{\text{r}}^{0\sim25\%}$ | $\overline{\text{r}}^{25\sim50\%}$ | $\overline{\text{r}}^{50\sim75\%}$ | $\overline{\text{r}}^{75\sim100\%}$ | $\overline{\text{r}}$ | Ent |
| HED+CE | 37.53 | 6.62 | 36.40 | 88.59 | 87.97 | 79.29 | 60.88 | 79.25 | 5.21 |
| AHED+CE | 39.11 | 6.71 | 38.16 | 88.87 | 86.74 | 78.16 | 60.71 | 78.62 | 5.11 |
| C-HED+CE | 36.93 | **6.96** | 39.52 | 90.01 | 88.11 | 84.03 | 69.89 | 82.52 | 5.17 |
| C-AHED+CE | 38.21 | 5.39 | 38.23 | 89.12 | 87.15 | 85.87 | 67.24 | 81.92 | 5.35 |
| C-HED+RL | 37.21 | 6.80 | **40.99** | 91.48 | **90.82** | 86.32 | 72.23 | 85.21 | **5.96** |
| C-AHED+RL | 40.05 | 6.85 | 40.53 | **92.30** | 90.34 | **85.82** | **72.80** | **85.31** | 5.73 |

## 4.7 Experimental results on entrained response generation

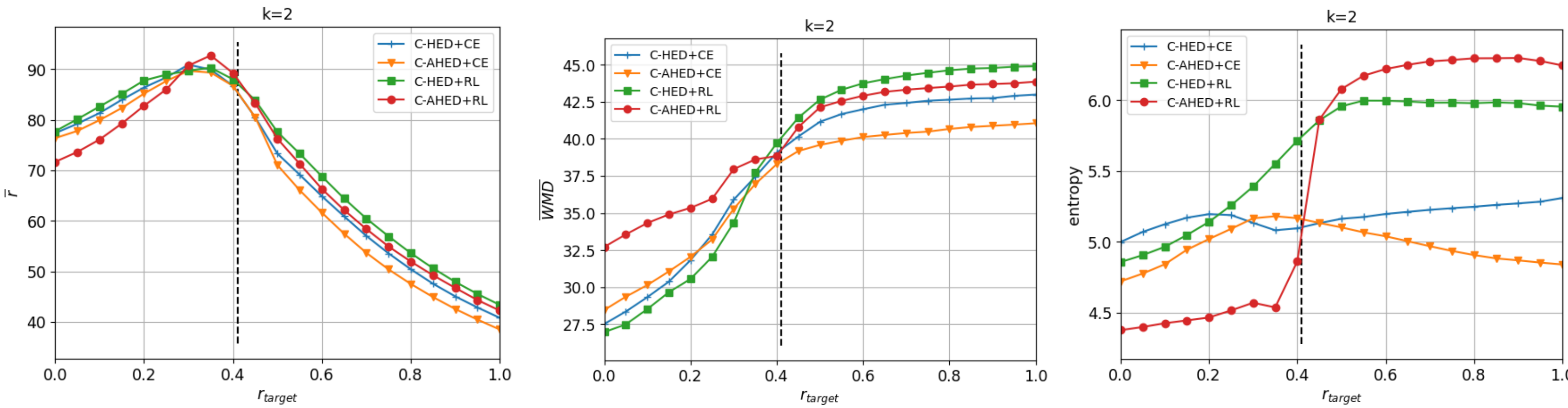
### 4.7.1 Automatic evaluation results

We show the automatic evaluation results of response generation models in Table 4.5. Our proposed models using the target entrainment degree as a condition showed improvement on $\bar{\mathrm{r}}$ from the baseline models and achieved comparable performance on other metrics. Our proposed models controlled the generation at a high level, based on the indicated entrainment degree. In particular, we confirmed a remarkable improvement in models that applied reinforcement learning (C-HED+RL and C-AHED+RL). C-AHED+RL showed the best performance in $k = 2, 3$, indicating that the attention mechanism for context works well when the model uses longer contexts. However, we still have a problem with generation performance $\bar{\mathrm{r}}^{75\sim100\%}$, which gives very high entrainment scores as a condition. In other words, generating highly entrained responses is more challenging. Furthermore, our proposed models based on reinforcement learning showed a consistent improvement of $\overline{\mathrm{WMD}}$ and Ent, and BLEU was comparable to the baseline models. For $k = 2$ and $k = 3$, there was no significant difference between them regarding the relevance of the responses. The model based on hierarchical encoders used in this experiment is not specialized to handle long contexts, thus setting a large $k$-value may not be worth the training cost of the model it incurs.

Then we traced the changes in the generation performance when we gave different fixed examples of $r_{\mathrm{target}}$ as a condition for the generation models instead of the oracle. The results are shown in Fig. 4.3. Here the vertical dashed line is the median of oracle $r_{\mathrm{target}}$. Our proposed models, which are optimized by reinforcement learning, showed consistent improvement compared with the other models. For $\bar{r}$, we confirmed the highest performance around the median of oracle $r_{\mathrm{target}}$. Lower scores on high $r_{\mathrm{target}}$ were probably caused by a lack of training samples of high $r_{\mathrm{target}}$. For $\overline{\mathrm{WMD}}$ and Ent, we confirmed increasing trends in both scores when we give a high $r_{\mathrm{target}}$. On the other hand, both scores are low in the range of low $r_{\mathrm{target}}$. This result was caused by dull responses, which have small diversity and little relevance to the references. In some cases, our models outperformed the results of giving the oracle conditions. This indicates that our models are robust even if the condition given to the model is different from the oracle.




(a) Performance comparison when $k = 1$




(b) Performance comparison when $k = 2$

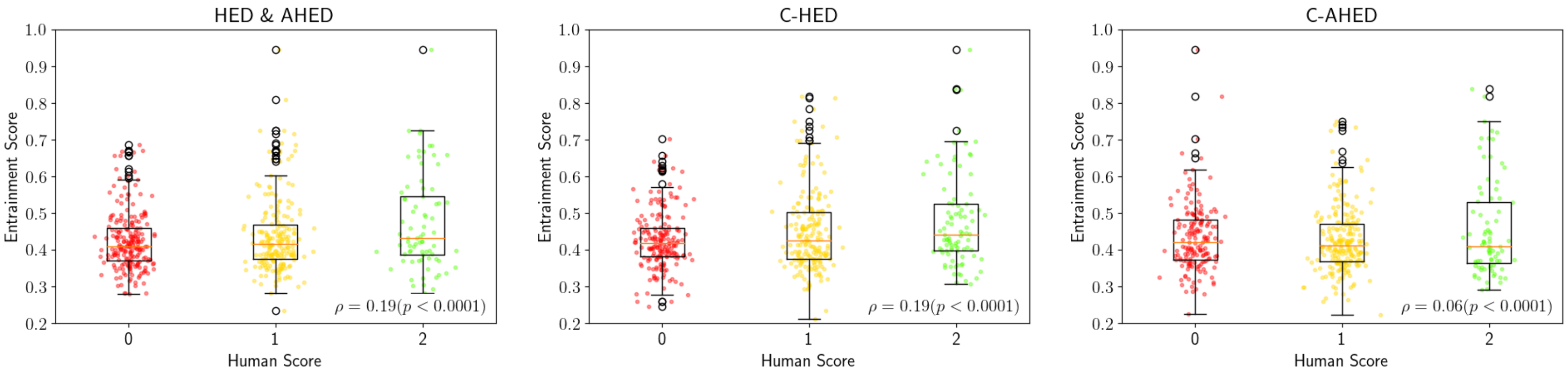
Figure 4.3: Changes in generation performance when given a fixed $r_{\text{target}}$

### 4.7.2 Human subjective evaluation results

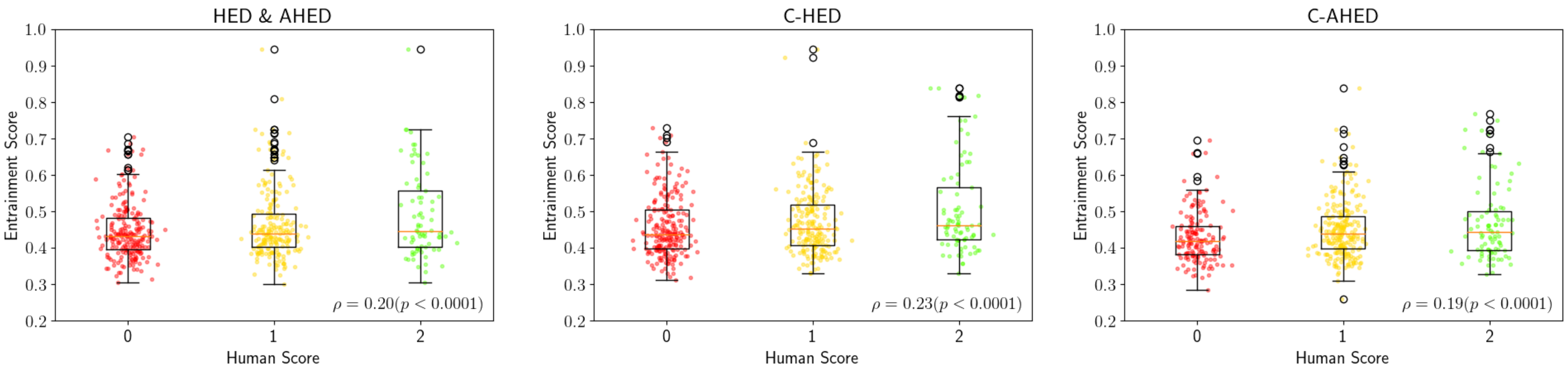
Table 4.6 shows the human evaluation results for the naturalness of the generated responses in each model. We used the oracle entrainment degrees as the given conditions. Our proposed models, based on reinforcement learning, generated a more acceptable response to the dialogue context than the baseline models under the oracle condition. C-AHED+RL ($k = 2$) showed the highest performance. However, C-HED+RL ($k = 2$) did not improve the score compared with C-HED+RL ($k = 1$). This indicates that C-HED model, which has no attention mechanism, can not take enough advantage of reward signals that considering the more past context.

We also evaluated the relationship between human evaluation scores and entrainment scores based on LID[**]. Fig. 4.4 shows the box-plots for human evaluation scores and entrainment scores of three models (no-conditioned models and C-HEDs, C-AHEDs). Here the horizontal axis indicates the human evaluation scores and the vertical axis indicates the entrainment scores. Note that C-HEDs and C-AHEDs are including both results of cases in +CE and +RL. We calculated the polyserial correlation $\rho$ [Olsson et al., 1982] between human scores and entrainment scores instead of Spearman's rank correlation since there is only a 3-point scale for human scores. We identified significant positive correlations between human scores and entrainment scores for all of the groups regardless of the $k$ settings. This result is consistent with the results of the preliminary analysis in Section 4.5. Note that we can not compare the magnitude of the correlations in each group. This is due to the correlations will be small for groups with a low frequency of score 0 since the nature of the evaluation score based on the 3-point scale.

[**]The LID was calculated based on (4.16).




(a) Box-plots for entrainment scores and human scores when $k = 1$




(b) Box-plots for entrainment scores and human scores when $k = 2$

Figure 4.4: Relationships between human scores and entrainment scores

Table 4.6: Frequency of each subjective evaluation score. Weighted-Avg is a weighted average by frequency of scores.

| Models | 2 | 1 | 0 | Weighted-Avg |
|---|---|---|---|---|
| HED+CE | 44 | 85 | 111 | 0.72 |
| AHED+CE | 33 | 119 | 88 | 0.77 |
| C-HED+CE ($k = 1$) | 43 | 88 | 109 | 0.73 |
| C-HED+CE ($k = 2$) | 40 | 103 | 97 | 0.76 |
| C-HED+RL ($k = 1$) | 49 | 105 | 86 | **0.85** |
| C-HED+RL ($k = 2$) | 45 | 106 | 89 | 0.82 |
| C-AHED+CE ($k = 1$) | 45 | 105 | 90 | 0.81 |
| C-AHED+CE ($k = 2$) | 44 | 110 | 86 | 0.83 |
| C-AHED+RL ($k = 1$) | 38 | 123 | 79 | 0.83 |
| C-AHED+RL ($k = 2$) | 47 | 126 | 67 | **0.92** |

Table 4.7 shows the generation examples of the compared models based on AHED in $k = 2$. Their naturalness seems at least comparable; even our proposed models generated more entrained responses. Note that it is difficult to find using the same words because our proposed method is based on WMD that calculates the semantic similarity in semantic spaces.

## 4.8 Conclusion

We proposed a neural conversation model that can control the entrainment degree of generated responses according to the given entrainment degree. We applied reinforcement learning to optimize our model for automatic entrainment scores that incorporate local interpersonal distance as a reward. Experimental results showed the entrainment scores correlated with human assessments in both human-human and human-machine dialogues in a chit-chat domain. Our proposed model also generated comparable or more natural responses than conventional models based on the minimization of cross-entropy loss, while the degree of entrainment of the generated responses is well controlled.

Although our method outperformed the existing method based on cross-entropy loss, the entrainment degree of generated response can still be improved. This is because our method does not have any mechanism to explicitly access the vocabulary used in the dialogue context on its decoding process. Hierarchical attention [Xing et al., 2018] or a copying mechanism [Gu et al., 2016] may explicitly solve this problem based on the word information in dialogue contexts.

The entrainment induced by our system was shown to be significantly correlated with the user assessment. However, it is not sufficient to conclude that response generation based on high entrainment always induces a high user assessment. Because our entrainment score only considers a minimal aspect of the entrainment phenomenon. A phenomenon where synchronization with each other in a dialogue, such as an entrainment, is strongly related to the attributes of the dialogue participants and the relationships between them [Giles et al., 1987]. Therefore, by focusing on personality and individuality, we will redefine entrainment from various theoretical levels (e.g., convergence and divergence) and achieve more fine-tuned control of response generation for building a more user-satisfied system. Of course, entrainment is not a universal phenomenon that can improve the dialogue system. Thus we will also be looking for ways to incorporate phenomena other than entrainment into the dialogue system.

Table 4.7: Generated responses from models given a dialogue history and a target entrainment degree: A and B denote system and user. One utterance of each speaker is given as a history in the example. Generation results of each model are shown after the context. Before generation results, given target entrainment degrees and rewards bestowed on generated responses are indicated by brackets.

| | |
|---|---|
| **A** | hi there ! how is your day ? |
| **B: AHED+CE** | i am doing well . how are you ? |
| **B: C-AHED+CE** | ($r_{\text{target}} = 0.51$, $r = 0.79$) i am doing well . just got back from the gym . how are you ?. |
| **B: C-AHED+RL** | ($r_{\text{target}} = 0.51$, $r = 0.98$) hello , i am good thanks . how are you ? |
| **B: REFERENCE** | ($r_{\text{target}} = 0.51$, $r = 1.0$) i am doing great out here |
| **A** | have you thought about finding another job ? |
| **B** | i want to leave the job but my parents make me stay |
| **A** | maybe if you found a better one before leaving . |
| **B** | yeah but i'll move out my parents soon . |
| **A** | very cool ! i loved it when my kids left . |
| **B** | i'll love it when i leave then i can do as i like |
| **A: AHED+CE** | i am sorry to hear that . do you have any pets ? |
| **A: C-AHED+CE** | ($r_{\text{target}} = 0.62$, $r = 0.72$) i am sorry to hear that . do you have any pets ? |
| **A: C-AHED+RL** | ($r_{\text{target}} = 0.62$, $r = 0.83$) i hope so too ! what else do you like to do in your spare time ? |
| **A: REFERENCE** | ($r_{\text{target}} = 0.62$, $r = 1.0$) it does not always work out that way |
| **A** | uh , i guess that is cool , so you like food ? |
| **B** | yup love it . i also survived cancer though . |
| **A** | bet that was pretty rough' |
| **B** | yes it was hard . i love to run . how about you ? |
| **A** | i spend a lot of time trying to cook , i'm not super good . |
| **B** | that is fine . i also like preparing for marathons . |
| **A: AHED+CE** | i love to cook , but i do not have a job , but i do not like it |
| **A: C-AHED+CE** | ($r_{\text{target}} = 0.73$, $r = 0.45$ ) i am sorry to hear that . do you have any pets ? |
| **A: C-AHED+RL** | ($r_{\text{target}} = 0.73$, $r = 0.55$) that is awesome , what kind of music do you like ? |
| **A: REFERENCE** | ($r_{\text{target}} = 0.73$, $r = 1.0$) is that super hard work ? |
| **A** | i used to paint and would need to stop taking my meds to do a big job . |
| **B** | i understand . i like the city . |
| **A** | i am currently a cleaning lady and do alot of work in the city . |
| **B** | that sounds fun and interesting . |
| **A** | it is . what do you do ? |
| **B** | i work on construction right now . |
| **A: AHED+CE** | that sounds like a lot of fun . do you have any hobbies ? |
| **A: C-AHED+CE** | ($r_{\text{target}} = 0.23$, $r = 0.95$) that sounds like a lot of fun . do you have any hobbies ? |
| **A: C-AHED+RL** | ($r_{\text{target}} = 0.23$, $r = 0.84$) that is cool . i am a professional person . |
| **A: REFERENCE** | ($r_{\text{target}} = 0.23$, $r = 1.0$) do you have any hobbies ? |
| **A** | that's great , i am a grad student |
| **B** | do you have any hobbies ? i really love sky diving , rollercoasters and cooking shows ! |
| **A** | kayaking and reading psychology books |
| **B** | i'd go kayaking but i'm really not good at swimming so i'd be scared |
| **A** | i love it , i'm an awesome swimmer |
| **B** | do you like hummus ? it is a great alternative as a vegan , i love it ! |
| **A: AHED+CE** | i do not have any pets , i do not have any pets |
| **A: C-AHED+CE** | ($r_{\text{target}} = 0.28$, $r = 0.93$) i like to go to the beach , do you have any hobbies ? |
| **A: C-AHED+RL** | ($r_{\text{target}} = 0.28$, $r = 0.95$) i like to go to the park and listen to music |
| **A: REFERENCE** | ($r_{\text{target}} = 0.28$, $r = 1.0$) not really a fan , i am a meats kind of guy |

# 5 Dialogue structure parsing on multi-floor dialogue

In this chapter, we studied how to automatically understanding how the intentions are expressed and contributed to practical conversation situations, such as multi-floor dialogue. We describe a dialogue structure parser that identifies the structures of multi-floor dialogues.

## 5.1 Introduction

In single-floor dialogues, each participant can access any of the dialogue contents. For example, two people talking face to face, or an online conference involving participants from different places is a single-floor dialogue because each participant can access all of the dialogue contents. By contrast, a multi-floor dialogue consists of multiple sets of dialogue participants, each conversing within their own floor, but also at least one multi-communicating member who is a participant of multiple floors and coordinating each to achieve a shared dialogue goal [Traum et al., 2018]. For example, in a restaurant, a server communicates with customers to take their orders in the dining room (one floor) and talks with other workers in the kitchen (another floor) who prepare the customer's food. All the participants work toward the joint goal of providing the customer with their desired meals, however in this case, only the server participates in both floors, conveying orders from customer to kitchen and perhaps information about item availability or speed from kitchen back to customers. Another example is in military units, where soldiers follow their commander's orders, which are decided at headquarters. Such situations are quite common in the real world, where we have different dialogue floors for decision-making and actions based on decisions.

Identifying aspects of multi-floor dialogue structure can be critical for building cooperative applications that have to participate in multi-floor dialogues, for example collaborative navigation robots [Lukin et al., 2018, Bonial et al., 2018]. However, most existing studies on dialogue structure parsing addressed only single-floor dialogues. There are standard annotation schemes for both dialogue acts [Bunt et al., 2012a] and discourse relations [Prasad and Bunt, 2015] in single-floor dialogues. Some proposed models have parsed the dialogue structure. However, these schemes do not fully address the issues of dialogue structure in multi-floor dialogues. A previous work proposed an annotation scheme of dialogue structure on multi-floor dialogues [Traum et al., 2018]. This scheme is based on two important aspects of dialogue structure: transaction units and the relations between utterances. A transaction unit clusters utterances from multiple participants and floors that contribute to achieving the initiating participant's intention. Relations link utterances to antecedents within the unit. However, there is no previous work on automatic dialogue structure parsing for multi-floor dialogue.

In this study, we propose a first neural dialogue structure parser for multi-floor dialogue structure. Our proposed parser has an attention mechanism to predict structure across different floors. In the following sections, we describe the dialogue structure parsing task on multi-floor dialogue, an annotation scheme, and our target domain (Section 5.2). We describe our proposed system based on the end-to-end approach, which automatically identifies the dialogue structure of multi-floor dialogues by recurrent neural networks. The definitions of transaction units, antecedents, and relation-types are closely related to each other. We applied the attention mechanism and multi-task learning to improve the overall performance of the dialogue structure parser considering their characteristics (Section 5.3). We experimentally evaluated the dialogue structure parsing performance of our model using automatic metrics that focus on micro- and meso-level structures [Traum and Nakatani, 1999] in dialogues (Section 5.4). Our proposed model using multi-task learning improved the overall performance compared to models trained on single-task settings (Section 5.5). Finally, we conclude by describing the performance of our proposed model and discussing possible future directions (Section 5.6).

Table 5.1: Dialogue example of multi-floor dialogue

| | Left Floor | | Right Floor | | Annotations | | |
|---|---|---|---|---|---|---|---|
| # | **Commander** | **DM → Commander** | **DM → RN** | **RN** | **TU** | **Ant** | **Rel** |
| 1 | move to where you see the first cone | | | | 1 | | |
| 2 | | I'm not sure which object you are referring to. Can you describe it in another way, using color or its location? | | | 1 | 1 | request-clarification |
| 3 | move to the cone on the right a red cone on the right | | | | 1 | 2 | clarification-repair |
| 4 | | | move to face the cone on the right | | 1 | 3 | translation-r |
| 5 | | executing... | | | 1 | 3 | ack-doing |
| 6 | take another picture | | | | 2 | | |
| 7 | | | | done | 1 | 4 | ack-done |
| 8 | | done | | | 1 | 7 | translation-l |
| 9 | | | image | | 2 | 6 | translation-r |
| 10 | | | | image sent | 2 | 9 | ack-done |
| 11 | | sent | | | 2 | 10 | translation-l |

## 5.2 Dialogue structure in multi-floor dialogue

For our initial investigations, we use a dataset of multi-floor dialogue structure, created as part of a long-term project to develop an autonomous robot [Marge et al., 2016,Lukin et al., 2018,Gervits et al., 2019], which is commanded by remote human participants. The robot is in an unfamiliar physical environment, where it performs object searches through natural language interaction. The dataset consists of "Wizard of Oz" dialogues where two wizards control the robot and communicate with the human commander. The dialogue manager wizard (DM) communicates directly with the commander in natural language and handles clarifications or misconceptions that might not be applicable given the environment and robot capabilities. A robot navigator wizard (RN) controls the robot with a joystick controller, but communicates only with the DM. There are thus two separate floors - one between commander and "robot" (actually the DM), and one between the two wizards. These floors are called "left" and "right", for convenience. Table 5.1 shows an example of an actual dialogue excerpt, including two floors and four distinct message streams. The commander gives its intention to the DM on their dialogue floor (left floor). The DM talks with the commander (when necessary) to clarify the commander's intention. After completely understanding the commander's intention, the DM moves to another dialogue floor (right floor) to transfer the commander's intention to the RN, which operates the robot based on the given intention and reports the result to the DM. The DM returns to the first floor to feedback the result to the commander. Note that the DM can communicate with any participants by moving among several dialogue floors to transfer the information as a multi-communicator [Reinsch Jr et al., 2008]; but the RN and the commander cannot directly communicate.

Previous work defined an annotation scheme for such multi-floor dialogues to specify their characteristics [Traum et al., 2018]. To capture the information update process of the dialogue participants, this scheme focused on the intentional structure [Grosz and Sidner, 1986], which consists of units of multiple consecutive utterances, and the relations between pairs of utterances within the unit. They defined an annotation scheme for (1) transaction units, (2) antecedents, and (3) relation-types, and the dataset includes human-annotated data. In this study, we explore a model that automatically identifies these structures. Below we describe the annotation scheme in [Traum et al., 2018].

### 5.2.1 Transaction unit

A transaction unit (TU) is a basic unit of intentional structure in a multi-floor interaction. It consists of the initial utterance that expresses the intention of the speakers and every subsequent utterance across all the floors to achieve the original speaker's intention. Each utterance belongs to a transaction unit, which is defined by a set of utterances. The "TU" column of Table 5.1 shows an numerical identifier for the unit which is the same for all utterances that are part of the TU.

In some cases, multiple transactions are "active" at the same time, in that they have been initiated but not terminated. For example, Table 5.1 shows a case where two transaction units are included in the dialogue: TU1 is about moving somewhere, while TU2 is about taking a picture. TU2 is initiated in utterance #6, before TU1 is completed in utterance #8. Both transactions are thus running in parallel during this part of the dialogue.

### 5.2.2 Antecedent and relation-type

In [Traum et al., 2018], relations are annotated between utterances in the same TU, using antecedents and relation-types. Any utterances after the first utterance in the transaction unit have antecedents, shown in the "Ant" column of Table 5.1, as the utterance ID of the antecedent utterance. Relation types are summarized in Table 5.2. These relations are categorized first as to whether they are from the same participant (expansions), from different participants in the same floor, or across floors. Each of these categories has a set of specific relations and in some cases sub-types. Relation types are indicated in the "Rel" column in Table 5.1.

The set of relations within a transaction define a tree structure, where the first utterance is the root node, which has no relation-type or antecedent annotations. In the example in Table 5.1, #1 and #6 are the root nodes of the two transaction units.

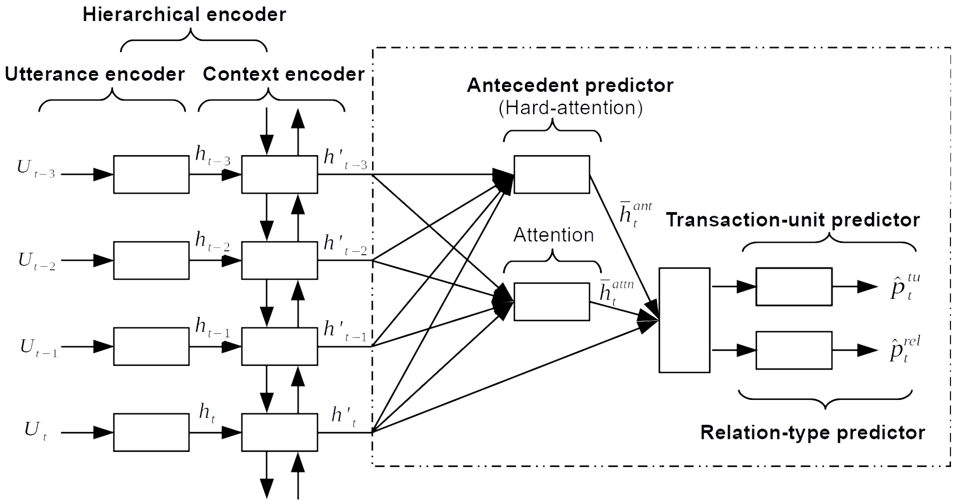



Figure 5.1: Overview of proposed neural dialogue structure parser

## 5.3 Neural dialogue structure parser for multi-floor dialogue

In this section, we introduce a neural dialogue structure parser for the annotation scheme proposed by [Traum et al., 2018]. A dialogue structure parser based on end-to-end neural networks improved the parsing performance more than legacy models using hand-crafted features [Afantenos et al., 2015, Shi and Huang, 2019]. Thus, we built an end-to-end neural dialogue structure parser model with available data and explored its limitations.

In our dialogue structure parsing task on multi-floor dialogues, three tasks are closely related: transaction units, antecedents, and relation-type identifications. We expect that multi-task learning will improve the overall parsing performance more than single models. The attention mechanism can explicitly represent their relations. Thus, our model is based on a recurrent neural network that has both soft- and hard-attention mechanisms with multi-task learning. The hard-attention employs the idea of MST parser based on biaffine attention [Dozat and Manning, 2016], a powerful previous approach to resolving relations between words, and captures the links between utterances beyond the relation within words. The soft attention has different parameters than hard attention, and it is expected to extract not only link relation of utterances but other implicit features that enhance the entire parsing results. Our proposed model (Fig. 5.1) mainly includes four networks:

- **Hierarchical encoder** has utterance and context encoders for encoding each dialogue context in different dialogue levels.
- **The antecedent predictor** estimates the antecedent that corresponds to each utterance.
- **The transaction-unit predictor** estimates the type of transaction boundaries of each utterance.
- **The relation-type predictor** estimates the relation-type of each utterance and its antecedent.

The transaction-unit and relation-type predictors share the prediction results of the antecedent predictor as attention weights, because their prediction results are related to the potential tree structures decided by the antecedent predictor model. Such a two-stage approach, which predicts the dependency structure of the utterances and its relation-types, resembles previous work [Shi and Huang, 2019]. However, that model targets single-floor dialogue structure parsing, and our model predicts the dialogue structure of multi-floor dialogues and clusters the utterances in different floors as one transaction unit.

## 5.3.1 Hierarchical encoder

Our hierarchical encoder consists of utterance and context encoders. The utterance encoder receives a word at each time step using forward and backward GRUs [Cho et al., 2014] to encode each utterance into a fixed-length vector:

$$\overrightarrow{h_{t,i}} = \overrightarrow{\mathrm{GRU}}_{\mathrm{utt}}(\overrightarrow{h_{t,i-1}}, \mathrm{Embedding}(w_{t,i})), \tag{5.1}$$

$$\overleftarrow{h_{t,i}} = \overleftarrow{\mathrm{GRU}}_{\mathrm{utt}}(\overleftarrow{h_{t,i+1}}, \mathrm{Embedding}(w_{t,i})), \tag{5.2}$$

$$h_{t,i} = [\overrightarrow{h_{t,i}}; \overleftarrow{h_{t,i}}], \tag{5.3}$$

$$h_t = \frac{1}{|U_t|} \sum_{i=1}^{|U_t|} h_{t,i}. \tag{5.4}$$

Here $t$ is the utterance numbers in the dialogue context and $i$ is the word order in the utterance. $h_{t,i}$ is the hidden vector calculated from each word $w_{t,i}$ and the hidden vector in previous time-step $h_{t,i-1}$ in utterance $U_t = \{w_{t,1}, w_{t,2}, \cdots, w_{t,N}\}$. Each word $w_{t,i}$ is converted to a fixed-length vector using an embedding layer before calculating the hidden vector. In each utterance, we added a special symbol, which indicates the types of floors, to prefixes and suffixes of utterance and trained the embedding rule as done with words.

In the context encoder, utterance vectors are input to encode the dialogue history to get context-level vector representation $h'_t$ for each utterance in the dialogue contexts:

$$\overrightarrow{h'_t} = \overrightarrow{\mathrm{GRU}}_{\mathrm{hist}}(\overrightarrow{h'_{t-1}}, h_t)), \tag{5.5}$$

$$\overleftarrow{h'_t} = \overleftarrow{\mathrm{GRU}}_{\mathrm{hist}}(\overleftarrow{h'_{t+1}}, h_t), \tag{5.6}$$

$$h'_t = [\overrightarrow{h'_t}; \overleftarrow{h'_t}]. \tag{5.7}$$

We introduce a soft-attention mechanism [Luong et al., 2015] for dialogue contexts to compute contextual representation $\bar{h}_t^{\mathrm{attn}}$ for each utterance $U_t$:

$$\mathrm{attention}(h'_{t-j}, h'_t) = h'^{T}_{t-j} W_{\mathrm{ant}} h'_t, \tag{5.8}$$

$$\alpha_j = \frac{\exp(\mathrm{attention}(h'_{t-j}, h'_t))}{\sum_{j=1}^{k} \exp(\mathrm{attention}(h'_{t-j}, h'_t)}, \tag{5.9}$$

$$\bar{h}_t^{\mathrm{attn}} = \sum_{j=1}^{k} \alpha_j \cdot h'_{t-j}. \tag{5.10}$$

Here $k$ is the number of previous utterances considered in the calculation of attention, $W_{\mathrm{attn}}$ is a trainable weight-matrix, and $\alpha_j \in [0, 1]^k$.

In addition, we introduce a hard-attention mechanism for explicitly considering the antecedent, which corresponds to each turn $t$:

$$\bar{h}_t^{\mathrm{ant}} = \sum_{j=1}^{k} \beta_j \cdot h'_{t-j} \tag{5.11}$$

Here $\beta_j$ takes 1 if utterance $U_{t-j}$ is the antecedent of utterance $U_t$ and 0 in other cases ($\beta_j \in \{0, 1\}^k$).

Attention vectors $\bar{h}_t^{\text{attn}}$ and $\bar{h}_t^{\text{ant}}$, which are calculated on the basis of the hard and soft-attention mechanisms, are combined:

$$\hat{h}_t^{\text{fc}} = \tanh(\text{Linear}_{\text{attn}}([\bar{h}_t^{\text{attn}}; \bar{h}_t^{\text{ant}}; h'_t])). \tag{5.12}$$

Here $\text{Linear}_{\text{attn}}$ is a linear transformation layer, which includes a bias term. $\hat{h}_t^{\text{fc}}$ is a shared vector for predicting the transaction units and relation-types. Note that gold antecedent $\beta$ is used in training; however, in the inference, the model uses predicted distribution of $\bar{h}_t^{\text{ant}}$ by the antecedent predictor.

### 5.3.2 Antecedent predictor

As shown in Table 5.1, each utterance has an annotation of the corresponding antecedent as its utterance ID (#). To predict the antecedents for each utterance $U_t$, we calculated the scores between each utterance and the contextual utterances:

$$\text{antecedent}(h'_{t-j}, h'_t) = h'^{T}_{t-j} W_{\text{ant}} h'_t, \tag{5.13}$$

$$\hat{\beta}_j = \frac{\exp(\text{antecedent}(h'_{t-j}, h'_t))}{\sum_{j=1}^{k} \exp(\text{antecedent}(h'_{t-j}, h'_t))}. \tag{5.14}$$

Here, $k$ is the number of preceding utterances that can be the antecedent, $W_{\text{ant}}$ is a trainable weight-matrix, and $\hat{\beta}_j \in [0, 1]^k$. By calculating the position of antecedent from the weights of attention, we can carry this knowledge forward to other predictions in the rater step: transaction-unit prediction and relation-types prediction.

We set the cross-entropy loss between predicted distribution $\hat{\beta}$ and actual antecedent label $\beta$ as a loss function that enforces that the contextual utterance has the highest score when it is the antecedent of $U_t$:

$$L_{t,\text{ant}} = -\sum_{j=1}^{k} \beta_j \log(\hat{\beta}_j). \tag{5.15}$$

Note that we also calculate the attention weight corresponding to the case where the utterance does not have any antecedent using the trainable vector and the hidden vector $h'_t$.

### 5.3.3 Transaction-unit predictor

We formulate the problem of transaction unit prediction as a sentence classification problem that determines the boundaries of the transaction units in dialogues. The transaction-unit predictor classifies each utterance into the following three classes:

- **Start**: the utterance is the beginning of a transaction unit.
- **Continue**: the utterance belongs to the same transaction unit as the previous utterance.
- **Other**: the utterance cannot be categorized into either of the above classes.

**Other** indicates that the utterance belongs to an already open transaction that is different from the one the previous utterance belongs to, such as utterance #7 and #9 in Table 5.1. We predict transaction boundaries using $\hat{h}_t^{\text{fc}}$, derived from the calculation results of soft and hard-attentions to the context:

$$\hat{p}_t^{\text{tu}} = \text{softmax}(\text{Linear}_{\text{tu\_pred}}(\hat{h}_t^{\text{fc}})). \tag{5.16}$$

Here $\text{Linear}_{\text{tu\_pred}}$ is a linear transformation layer that includes a bias term, and $\hat{p}_t^{\text{tu}}$ is the predicted distribution of the transaction boundaries.

We used the cross-entropy loss as the loss function:

$$L_{t,\text{tu}} = -\sum_{j=1}^{|p_t^{\text{tu}}|} p_j^{\text{tu}} \log(\hat{p}_{t,j}^{tu}). \tag{5.17}$$

Here $p_t^{\text{tu}}$ is a three-dimensional vector corresponding to the type of target transaction boundaries.

### 5.3.4 Relation-type predictor

We used $\hat{h}_t^{\text{fc}}$ as well as the transaction-unit predictor to predict the relation-type of each utterance with its antecedent:

$$\hat{p}_t^{\text{rel}} = \text{softmax}(\text{Linear}_{\text{rel\_pred}}(\hat{h}_t^{\text{fc}})). \tag{5.18}$$

Here $\text{Linear}_{\text{rel\_pred}}$ is a linear transformation layer that includes the bias term and $\hat{p}_t^{\text{rel}}$ is the predicted distribution of the relation-types.

We used the cross-entropy loss for the training:

$$L_{t,\text{rel}} = -\sum_{j=1}^{|p_j^{\text{rel}}|} p_j^{\text{rel}} \log(\hat{p}_{t,j}^{rel}). \tag{5.19}$$

Here $p_t^{\text{rel}}$ is a vector, whose dimensions correspond to a relation label defined in Table 5.2.

### 5.3.5 Objective function

We have to optimize the above three models not only to a single model but also to the other two models because these tasks are closely related. In this study, we introduce a multi-task loss, which combines each prediction loss of the antecedent, the transaction-unit, and the relation-type predictor. In multi-task learning, we interpolate the loss functions of three tasks:

$$L = \frac{1}{N} \sum_{t=1}^{N} (\gamma_{\text{ant}} L_{t,\text{ant}} + \gamma_{\text{tu}} L_{t,\text{tu}} + \gamma_{\text{rel}} L_{t,\text{rel}}). \tag{5.20}$$

Here $N$ is the dialogue length. $\gamma_{\text{ant}}$, $\gamma_{\text{tu}}$, and $\gamma_{\text{rel}}$ are the weights for adjusting the importance of each predictor in the loss calculation.

## 5.4 Experimental settings

In our experiment, we evaluated the dialogue structure parsing performance of our proposed model. In this section, we describe the dataset for the training and evaluation, the setting of the model training, and the evaluation metrics.

### 5.4.1 Dataset

We used a dataset [Traum et al., 2018] that contains Exp. 1 and Exp. 2 data*. The dialogues were annotated based on a previously described scheme [Traum

*The annotation for Exp. 3 is still in progress.

et al., 2018], which was specifically designed to handle multiple conversational floors. As shown in Table 5.3, these dialogue data consist of 48 dialogues (1829 transactions) executed by several different commanders.

To evaluate the parsing performance of the proposed model, we randomly divided all of the dialogues in Exp. 1 and Exp. 2 into six subsets and applied double cross-validation [Mosier, 1951]. We used a single subset for validation and a test-set for each, and the remaining subset was used as training data. We evaluated every possible combination of training, validation, and test-set and the final performance by a majority vote on the prediction results of the models, which share the same test-set.

Table 5.2: Relation-types in a multi-floor dialogue

| Type | Sub-types |
|---|---|
| **Expansions** | relate utterances that are produced by the same participant within the same floor. |
| | continue |
| | link-next |
| | correction |
| | summarization |
| **Responses** | relate utterances by different participants within the same floor. |
| | acknowledgment |
| |     done |
| |     doing |
| |     wilco |
| |     understand |
| |     try |
| |     unsure |
| |     can't |
| | clarification |
| |     req-clar |
| |     clar-repair |
| |     missing info |
| |     nack |
| |     repeat |
| | processing |
| | question-response |
| |     answer |
| |     non-answer |
| | other |
| | 3rd turn feedback |
| | reciprocal response |
| **Translations** | relate utterances in different floors. |
| | transalation-l |
| | transalation-r |
| | comment |
| | quotation |

Table 5.3: Numbers of dialogues, utterances, and transactions

| | Dialogues | Utterances | Transactions |
|---|---|---|---|
| Exp. 1 | 24 | 4527 | 780 |
| Exp. 2 | 24 | 6994 | 1049 |

Table 5.4: Prediction performances of transaction units, antecedents, and relation-types

| Models | TU | | | | Ant | | | | Rel | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\overline{\text{Prec.}}$ | $\overline{\text{Rec.}}$ | $\overline{\text{F1}}$ | TuAcc | $\overline{\text{Prec.}}$ | $\overline{\text{Rec.}}$ | $\overline{\text{F1}}$ | TreeAcc | $\overline{\text{Prec.}}$ | $\overline{\text{Rec.}}$ | $\overline{\text{F1}}$ | TreeAcc w/ rel |
| **Majority** | 55.71 | 74.64 | 63.80 | - | 23.59 | 48.57 | 31.76 | - | 8.54 | 29.22 | 13.21 | - |
| **Single-Online** | 95.43 | 95.46 | 95.44 | 81.19 | 93.92 | 90.89 | 92.34 | 68.12 | 92.38 | 92.84 | 92.53 | 63.80 |
| **Single-Online w/o floor emb** | 94.41 | 94.46 | 94.43 | 77.41 | 93.61 | 90.18 | 91.81 | 67.57 | 91.70 | 92.09 | 91.79 | 60.30 |
| **Multi-Online** | 95.99 | 95.99 | 95.99 | 84.25 | 93.93 | 90.84 | 92.33 | 70.09 | 93.69 | 94.11 | 93.80 | 66.81 |
| | (96.26) | (96.27) | (96.26) | (85.34) | - | - | - | - | (94.75) | (94.94) | (94.77) | (67.74) |
| **Multi-Online w/o floor emb** | 94.62 | 94.63 | 94.62 | 78.18 | 92.58 | 89.20 | 90.81 | 66.86 | 91.24 | 91.97 | 91.58 | 63.31 |
| **Single-Offline** | 95.31 | 95.35 | 95.33 | 81.46 | 94.26 | 90.70 | 92.40 | 68.83 | 94.86 | 93.22 | 92.91 | 64.62 |
| **Single-Offline w/o floor emb** | 94.54 | 94.62 | 94.57 | 77.96 | 91.20 | 91.60 | 91.31 | 65.99 | 92.37 | 88.68 | 90.43 | 62.38 |
| **Multi-Offline** | **96.05** | **96.07** | **96.06** | **84.52** | **94.58** | **91.95** | **93.21** | **71.35** | **93.75** | **94.26** | **93.90** | **69.05** |
| | (96.30) | (96.31) | (96.30) | (85.51) | - | - | - | - | (94.68) | (94.95) | (94.69) | (69.92) |
| **Multi-Offline w/o floor emb** | 94.92 | 94.96 | 94.93 | 78.73 | 93.63 | 90.68 | 92.08 | 69.21 | 92.14 | 92.41 | 92.16 | 65.39 |

### 5.4.2 Model settings

We evaluated the dialogue structure parsing performance of the proposed model in multi-floor dialogues by comparing the dialogue structure parsing performances of the proposed model with the multi-task loss (**Multi**) and the models individually trained for each task (**Single**). We also compared the cases based on both the **Offline** and **Online** models. The proposed model described in Section 3 uses bi-directional GRUs in the context encoder to make predictions for each utterance $U_t$; this means the model cannot start parsing during the dialogue. We call this setting **Offline**. In contrast, we also considered a model that only uses previous contexts without subsequent contexts in the prediction for each utterance $U_t$. We call this setting **Online**. The online model is important for real-time dialogue robot processing, which can only use the observed information based on the interaction sequence. We built the online model only using forward-GRUs instead of bidirectional-GRUs in the context encoder. We also compared the case where the model does or does not consider the floor information when calculating the embedding vectors of utterances. In other words, we compared the case where a multi-floor dialogue is considered as just a multi-party dialogue with the case where the floor structure of the dialogue is considered.

We used the same hyper-parameter settings in each model. The vocabulary size was 500, the word embedding size was 100, and the hidden size was 300. We used byte pair encoding (BPE) for tokenization [Sennrich et al., 2016]. In training, we used a mini-batch size of 64 and an Adam optimizer [Kingma and Ba, 2014] with a learning rate of 1e-4. We set $\gamma_{\mathrm{ant}}$, $\gamma_{\mathrm{tu}}$, and $\gamma_{\mathrm{rel}}$ to 1. In the relation-type prediction, we integrated the 'acknowledgement," "clarification," and "question-response" sub-types into these classes because some sub-types rarely appeared in the dataset. In addition, we defined a label where utterance has no antecedent, as well as relation-types (#1 and #6 in Table 5.1).

### 5.4.3 Evaluation metrics

We defined the micro and meso-level evaluation metrics for our dialogue structure parsing task. For the micro-level evaluation, we defined the label prediction performances of the antecedents, the transaction units, and the relation-types by precision (Prec.), recall (Rec.), and F1. Note that we took the relative position of each utterance from its antecedent as a label to compute the metrics when evaluating the antecedent prediction performance. In other words, we compared the difference between the position of predicted antecedents and actual antecedents. We also introduced metrics for the meso-level structure [Traum and Nakatani, 1999] in dialogues to evaluate the consistency of the parsing results. We used the following three metrics:

- **TuAcc** is the ratio of the transaction units that perfectly predicted the transaction boundaries for each utterance within the transaction unit.
- **TreeAcc** is the ratio of the transaction units that perfectly predicted the antecedents for each utterance within the transaction unit.
- **TreeAcc w/ rel** is the ratio of the transaction units that perfectly predicted the antecedents and the relation-types for each utterance within the transaction unit.

Note that the meso-level metrics are stricter than the micro-level metrics, which judge the prediction result of each utterance.

## 5.5 Experimental results

Table 5.4 shows the performances of each dialogue structure parser. Here **Single** denotes a case where the transaction unit, the antecedent, and the relation-type predictors were individually trained. **Multi** denotes a case where these models were trained with multi-task learning loss. In addition, **Offline** indicates that the model used the dialogue entirely for parsing, and **Online** indicates that the model used only the preceding context of each utterance. **Majority** denotes a case where the always predicts the most frequent label. **w/o floor emb** indicates that the embedding vectors of the floor is not used in the model. $\overline{\text{Prec.}}$, $\overline{\text{Rec.}}$, and $\overline{\text{F1}}$ are the weighted averages† of the precision, recall, and F1 scores of the predicted labels. The brackets are the prediction results where the oracle antecedent was fed into the model.

†We calculated the weighted averages based on the label frequencies.

Table 5.5: Transaction unit prediction performance of Multi-Offline model: The brackets show the differences from Multi-Online model.

| Tu-Label | Precision | Recall | F1 | Count |
|---|---|---|---|---|
| Start | 94.66 | 93.06 | 93.85 (+0.23) | 1829 |
| Continue | 97.50 | 98.00 | 97.75 (+0.03) | 8599 |
| Other | 86.94 | 85.91 | 86.42 (+0.12) | 1093 |
| Weighted-Avg | 96.05 | 96.07 | 96.06 (+0.07) | 11521 |

The result shows that the dialogue structure parsing performance (transaction unit prediction, antecedent prediction, and relation-types prediction) of the **Offline** models have slightly improved from the **Online** models. This result indicates that subsequent contexts are useful for predicting labels for each utterance, but we have enough prediction accuracy without using the subsequent context. When the **Multi** models use oracle antecedents to predict the transaction units and the relation-type, we can further improve the performance of dialogue structure parsing. When the floor information was not used, the dialogue structure parsing performance decreased significantly. This implies that multi-floor dialogue structure parsing does not work properly with the dialogue structure parser for just multi-party dialogue.

Table 5.5 shows the results of the transaction unit prediction in **Multi-Offline** and **Multi-Online** models, which were the best models in our experiment. We confirmed that the model achieved over 85% F1 for all labels (types of transaction boundary). However, there is still a problem in predicting “Other” (when the utterance belongs to a different transaction than the previous utterance, but it is not the start of a new transaction). Our models decided labels with the highest prediction probability for each utterance; however, we did not take into account the consistency of the prediction results in the sequence of dialogue. To solve this problem, we can introduce a model that takes into account information about the entire prediction results, such as Conditional Random Field (CRF) [Lafferty et al., 2001] for further improvements.

Table 5.6 shows the results of the antecedent prediction in the **Multi-Offline** and **Multi-Online** models. Here each label indicates the relative position from each utterance to its antecedent. Note that this table only shows the prediction results by considering a maximum of 10 previous utterances. We excluded

Table 5.6: Antecedent prediction performance of Multi-Offline model: The brackets show the differences from Multi-Online model.

| Position | Precision | Recall | F1 | Count |
|---|---|---|---|---|
| -10 | 58.33 | 66.67 | 62.22 (-4.45) | 21 |
| -9 | 73.08 | 57.58 | 64.41 (+4.40) | 33 |
| -8 | 82.50 | 63.46 | 71.74 (-2.17) | 52 |
| -7 | 80.77 | 74.12 | 77.30 (-2.20) | 85 |
| -6 | 90.48 | 85.39 | 87.86 (-0.57) | 178 |
| -5 | 81.31 | 64.44 | 71.90 (+1.21) | 135 |
| -4 | 91.87 | 85.32 | 88.48 (+2.27) | 477 |
| -3 | 94.98 | 87.00 | 90.82 (+1.32) | 1023 |
| -2 | 94.84 | 93.06 | 93.94 (+0.79) | 2509 |
| -1 | 95.99 | 95.47 | 95.73 (+0.82) | 4262 |
| Weighted-Avg | 94.58 | 91.15 | 93.21 (+0.88) | 8775 |

a few cases when the antecedent of the utterance is not included in the ten previous utterances, and the utterance has no antecedent. Our models can predict antecedents with high performance when the relative position was not distant. On the other hand, the prediction performance was below 80% when the relative positions were distant (greater than five in absolute). This result suggests the difficulty of addressing long-term dependency in dialogues. In addition, our model ignores the consistency of the tree structure associated with the predicted antecedents. The search for dialogue structures using dynamic programming probably has the potential to improve the performance of our model.

Table 5.7 shows the results of the relation-type predictions in the **Multi-Offline** and **Multi-Online** models. Our models showed higher F1 scores in frequent relation-types, including when the utterance has no antecedent. There is still a challenge in predicting low-frequent relation-types due to the lack of training data. Ongoing annotation work [Traum et al., 2018] with additional data may remedy this problem. We also need to look at ways to deal with these unbalanced labels.

As shown in Table 5.5, 5.6, and 5.7, there was actually little difference in performance between the **Multi-Offline** and **Multi-Online** models. Because they both use a greedy method of predicting dialogue structure, adopting the label with the highest prediction probability. In order to increase the benefits of

Table 5.7: Relation-type prediction performance of Multi-Offline model: The brackets show the differences from Multi-Online model.

| Relation-Type | Precision | Recall | F1 | Count |
|---|---|---|---|---|
| Expansions | | | | |
| -continue | 90.90 | 88.90 | 8989 (+1.74) | 955 |
| -link-next | 99.37 | 99.69 | 9953 (+0.00) | 318 |
| -correction | 40.00 | 11.11 | 1739 (-16.65) | 36 |
| -summarization | 0.00 | 0.00 | 0.00 (+0.00) | 13 |
| Responses | | | | |
| -acknowledgement | 97.29 | 97.06 | 97.17 (+0.19) | 3366 |
| -clarification | 79.51 | 85.00 | 82.16 (+1.18) | 420 |
| -processing | 0.00 | 99.57 | 99.78 (+0.21) | 233 |
| -question-answer | 57.05 | 51.74 | 54.27 (-1.70) | 172 |
| -other | 33.33 | 6.06 | 10.26 (-12.47) | 33 |
| -3rd-turn-feedback | 50.00 | 0.40 | 7.41 (+0.96) | 25 |
| -reciprocal-response | 0.00 | 0.00 | 0.00 (+0.00) | 5 |
| Translations | | | | |
| -l | 95.93 | 98.14 | 97.03 (+0.06) | 1563 |
| -r | 98.30 | 98.40 | 98.35 (+0.07) | 1942 |
| -comment | 40.00 | 38.10 | 39.02 (+18.51) | 21 |
| No-antecedent | 91.85 | 94.63 | 93.22 (-0.47) | 2419 |
| Weighted-Avg | 93.75 | 94.26 | 93.90 (+0.10) | 11521 |

the offline model, it is important to perform global optimization to improve the consistency of the dialogue structure parsing results. In addition, when inferring the dialogue structure in the online model, it is also important to utilize the parsing results in the previous turns whenever a new utterance is observed. We should improve the model we have built according to the intended use of the dialogue structure parsing model.

Finally, in Table 5.8, 5.9, we show an example of dialogue structure parsing on a fragment of multi-floor dialogue. Note that we displayed the correct labels in brackets when the label was incorrectly predicted, and "#" corresponds to cases where the utterance does not have the antecedent. The first example shows that the model accurately predicts all the transaction boundaries, antecedents, and relation-types, even if transactions were interleaved. However, the second example is including error predictions of transaction boundaries. In this example, there are only two TUs, but the model has determined that the utterance has three TUs. Note that, even if we assume the prediction of TU at #8 is correct, the prediction at #11 is still not correct. When such confusion occurs, the error extends beyond one utterance to multiple utterances. In many cases, delays in communication and differences in the quality of annotations between Exp.1 and 2 often confuse predictions.

Table 5.8: Examples of the dialogue structure parsing on multi-floor dialogue (correct case)

| | Left Floor | | Right Floor | | Prediction | | |
|---|---|---|---|---|---|---|---|
| **#** | **Commander** | **DM → Commander** | **DM → RN** | **RN** | **TU** | **Ant** | **Rel** |
| 1 | turn right twenty degrees | | | | Start | # | # |
| 2 | | | turn right 20 | | Continue | 1 | translation-r |
| 3 | | executing ... | | | Continue | 1 | response-ack. |
| 4 | | | image | | Continue | 1 | translation-r |
| 5 | | | | done image sent | Continue | 4 | response-ack. |
| 6 | go forward fifteen feet | | | | Start | # | # |
| 7 | | sent | | | Other | 5 | translation-l |
| 8 | and go through door on right | | | | Other | 6 | expansion-cont. |
| 9 | | | move forward about 15 feet , going through door on right , image | | Continue | 8 | translation-r |
| 10 | | executing ... | | | Continue | 8 | response-ack. |

Table 5.9: Examples of the dialogue structure parsing on multi-floor dialogue (error case)

| | Left Floor | | Right Floor | | Prediction | | |
|---|---|---|---|---|---|---|---|
| **#** | **Commander** | **DM → Commander** | **DM → RN** | **RN** | **TU** | **Ant** | **Rel** |
| 1 | take a picture | | | | Start | # | # |
| 2 | | | image | | Continue | 1 | translation-r |
| 3 | | | | image sent | Continue | 2 | response-ack. |
| 4 | | sent | | | Continue | 3 | translation-l |
| 5 | turn left ninety degrees | | | | Start | # | # |
| 6 | | | turn left 90 | | Continue | 5 | translation-r |
| 7 | | executing ... | | | Continue | 5 | response-ack. |
| 8 | take a picture after each command | | | | Start (**Continue**) | # (**5**) | # (**expansion-cont.**) |
| 9 | | | | done | Other (**Continue**) | 6 | response-ack. |
| 10 | | | take pic after each command | | Other (**Continue**) | 8 | translation-r |
| 11 | | | image | | Other (**Continue**) | 8 | translation-r |
| 12 | | | | image sent | Continue | 11 | response-ack. |
| 13 | | sent | | | Continue | 12 | translation-l |

## 5.6 Conclusion

We built a neural dialogue structure parser with an attention mechanism that applies multi-task learning to automatically identify the dialogue structure of multi-floor dialogues. The experimental results showed that our proposed model improved the identification performance on all tasks compared to the model trained on single task settings. However, problems remain with the performance of the dialogue structure identification due to the lack of training data, especially for rare labels. To prevent this problem, we will consider pre-training and the transfer learning of models using existing dialogue corpora and discourse-relation datasets. We also explore the possibility of introducing powerful models of similar tasks related for predicting tree-structure in a document, such as a dependency parsing [Nivre, 2010] and discourse parsing based on rhetorical structure theory [Webber et al., 2012, Mann and Thompson, 1988].

This study has developed the first baseline model for automatic identification of dialogue structure on multi-floor dialogues. It has a potential for applying to the automatic annotation of dialogue structure on multi-floor dialogues and encourages the development of a dialogue manager and robot navigator on multi-floor settings. We would also improve communication protocols via natural language by analyzing how intentions of participants are communicated on different dialogue floors using our dialogue structure parser.

# 6 Conclusion

## 6.1 Summary

Neural conversation models are end-to-end schemes that generate system responses from user utterances. However, various challenges remain to control response generation. Conditional neural conversation models offer a promising approach to this problem, controlling response generation by conditioning the network on specific intentions considering conversation structures and conversation phenomena. This dissertation focused on three conditional neural conversation model problems.

The first study (Chapter 3) focused on response generation controllability in conditional neural conversation models. We considered a conditional neural conversation model where model responses could be controlled by specific intentions considering conversation structure, such as dialogue acts. The system can effectively generate consistent responses towards a dialogue goal using intentions that consider conversational structure. However, current conditional neural conversation models do not sufficiently guarantee high-quality response generation representing the given intention. Therefore, we proposed a conditional neural conversation model with a new label-aware objective function that promotes highly discriminative response generation based on the given intention while maintaining generated response naturalness. Experimental results confirmed the proposed model generated promising responses in terms of controllability and naturalness compared with strong conventional models.

The second study (Chapter 4) focused on incorporating entrainment, an attractive human phenomenon, into neural conversation models. Entrainment is a well-known conversational phenomenon where conversation participants mutually synchronize regarding various aspects, is thought to be closely related to human-human conversation quality. We first analysed relationships between conversation quality and entrainment using an automatic entrainment evaluation measure, and showed that entrainment improved participant satisfaction in human-human and human-machine conversations. Therefore, we subsequently proposed a conditional neural conversation model to control generation using a given entrainment degree as the intention for response generation. Experimental results confirmed that the proposed entrainable neural conversation model generated comparable or more natural responses than current conventional models, and could satisfactorily control generated response entrainment.

The third study (Chapter 5) focused on automatically understanding how intentions were expressed and contributed to practical conversation situations. We concentrated on multi-floor dialogue, i.e., dialogues that span multiple conversational floors. Most current dialogue systems, including neural conversation models, do not consider multi-floor dialogues. Expanding the research scope will contribute to building autonomous systems to solve real-world problems using multi-floor dialogues. We first automatically identified how participants would proceed with a dialogue in domains such as urban search and rescue or military reconnaissance. As a first step, we proposed a baseline model that automatically identifies multi-floor dialogue structure based on multi-task learning and an attention mechanism. We showed experimentally that the proposed model achieved promising identification performance for dialogue structure and discussed its limitations.

## 6.2 Perspectives

This dissertation focused on three conditional neural conversation model problems. However, much remains to be addressed to improve further and to make the proposed models more attractive.

Neural conversation models discussed in Chapters 3 and 4 have been trained on limited datasets, leaving considerable room for improvement in response quality. Human subjective evaluations showed that generated responses around of about 30% were not meaningful or acceptable to dialogue context. This problem becomes more apparent when user and system converse through multiple turns. Language models trained on massive language resources have been shown as promising approaches for various NLP tasks to address data sparsity. Future studies should investigate using such pre-trained language models for the proposed models. Furthermore, since the models proposed in Chapters 3 and 4 apply optimization at the turn level, they did not consider the effects of unnatural conversations that accumulate when the model talks to the user for multiple turns. Therefore, we also need to consider global optimization methods that maximize response generation performance when continuing a conversation with a user from start to finish.

We need a neural conversation model to make effective use of context. Neural conversation models used in Chapters 3 and 4 were insufficiently capable of considering the long-term context. We applied an objective function considering entrainment and an attention mechanism in Chapter 4, to leverage dialogue context on neural conversation models. However, it remains challenging to generate responses that incorporate many words from the dialogue context because current methods do not have mechanisms to explicitly access the vocabulary used in the dialogue context during decoding. Hierarchical attention or a copying mechanism could explicitly solve this problem based on word information in dialogue contexts.

We should consider controllable neural conversation models integrating entrainment and various intentions. The approach to control response generation entrainment described in Chapter 4 focused on lexical choice in the dialogue. However, lexical choice is only one entrainment factor, with other factors including syntax, style, acoustic prosody, turn-taking, and dialogue act commonly employed in human-human dialogue. Thus, we need to understand how various factors are used in dialogue and how they affect dialogue quality, then build neural conversation models that integrate different control factors. By theoretically integrating the various types of intentions, we ultimately aim to develop a sophisticated system that does not violate human conversation principles such as Grice's maxims. We believe that neural conversation models can develop more human-like behavior by reproducing human conversational phenomena, including entrainment.

We should extend the research basis to incorporate real-world multi-floor dialogue. The dataset we used in Chapter 5 incorporated two floors and three participants, including one multi-communicator, which was the minimum requirement for multi-floor dialogue, and the proposed model had easily identifiable dialogue structure since only simple commands were used in the dataset. Therefore, we need to analyze more difficult dialogues with more floors and participants to develop a high-level decision-making model that addresses multi-floor dialogue, and connecting various dialogue systems, including conditional neural conversation models.

We will try to develop an overall system in the long term to address and/or integrate these identified issues.

# Acknowledgements

I would like to thank Professor Satoshi Nakamura for his continuous and enthusiastic supervision during my master's and doctoral course five years. His kind advice and guidance, which improved my ability as a researcher. I have been and will always be looking up to him as a great teacher and researcher.

I would like to thank Professor Taro Watanabe as a member of my doctoral thesis committee for reviewing this thesis. He pointed out ambiguous writings of my thesis, especially on technical viewpoints.

I would like to thank Dr. Koichiro Yoshino of the Institute of Physical and Chemical Research (RIKEN) for his supervision during the five years of my master's and doctoral course. He gave me advice not only on my research but also on my future carrier. I think that I was able to obtain my doctoral degree due to his supervision and support.

I would like to thank Professor David Traum of the USC Institute for Creative Technologies (USC-ICT) for agreeing to participate in my thesis committee and supervising my research internship at USC-ICT. Through discussion with him, I could get inspiration for the direction of my research.

I would like to thank Dr. Masahiro Mizukami of NTT Communication Science Laboratories for his supervision during my research internship at NTT. Through my internship at NTT, I was able to conduct the part of the study that made up my doctoral thesis.

I would like to thank Associate Professor Katsuhito Sudoh, Associate Professor Sakriani Sakti, Assistant Professor Hiroki Tanaka, and Assistant Professor Seitaro Shinagawa for the support, discussion, and constructive feedback during an internal research meeting.

I would like to thank Assistant Manami Matsuda, Miho Hayashi, and Technical Staff Hideko Habe for their valuable administrative and technical support. I would also like to thank all colleagues who worked hard on research together at NAIST.

Finally, I would like to thank my family for their support. Their encouragement always supports me in difficult times.

# References

[Adam et al., 2020] Adam, M., Wessel, M., and Benlian, A. (2020). Ai-based chatbots in customer service and their effects on user compliance. *Electronic Markets*, pages 1–19.

[Afantenos et al., 2015] Afantenos, S., Kow, E., Asher, N., and Perret, J. (2015). Discourse parsing for multi-party chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 928–937, Lisbon, Portugal. Association for Computational Linguistics.

[Alston, 2000] Alston, W. P. (2000). *Illocutionary acts and sentence meaning.* Cornell University Press.

[Arimoto et al., 2019] Arimoto, T., Sugiyama, H., Mizukami, M., Narimatsu, H., and Higashinaka, R. (2019). Analysis of conversation topic between first-met speakers over multiple text-chats. *SIG-SLUD*, 5(03):66–71.

[Asai et al., 2020] Asai, S., Yoshino, K., Shinagawa, S., Sakti, S., and Nakamura, S. (2020). Emotional speech corpus for persuasive dialogue system. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 491–497.

[Austin, 1975] Austin, J. L. (1975). *How to do things with words*, volume 88. Oxford university press.

[Balakrishnan et al., 2019] Balakrishnan, A., Demberg, V., Khatri, C., Rastogi, A., Scott, D., Walker, M., and White, M. (2019). Proceedings of the 1st workshop on discourse structure in neural nlg. In *Proceedings of the 1st Workshop on Discourse Structure in Neural NLG.*

[Ball and Breese, 2000] Ball, G. and Breese, J. (2000). Emotion and personality in a conversational agent. *Embodied conversational agents*, 189.

[Bengio et al., 1994] Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166.

[Beňuš et al., 2014] Beňuš, v., Gravano, A., Levitan, R., Levitan, S. I., Willson, L., and Hirschberg, J. (2014). Entrainment, dominance and alliance in supreme court hearings. *Know.-Based Syst.*, 71(1):3–14.

[Bobrow et al., 1977] Bobrow, D. G., Kaplan, R. M., Kay, M., Norman, D. A., Thompson, H., and Winograd, T. (1977). Gus, a frame-driven dialog system. *Artificial intelligence*, 8(2):155–173.

[Bonial et al., 2018] Bonial, C., Lukin, S. M., Foots, A., Henry, C., Marge, M., Pollard, K. A., Artstein, R., Traum, D., and Voss, C. R. (2018). Human-robot dialogue and collaboration in search and navigation. In *Proceedings of the Annotation, Recognition and Evaluation of Actions (AREA) Workshop of the 2018 Language Resources and Evaluation Conference (LREC).*

[Boyer et al., 2010] Boyer, K., Ha, E. Y., Phillips, R., Wallis, M., Vouk, M., and Lester, J. (2010). Dialogue act modeling in a complex task-oriented domain. In *Proceedings of the SIGDIAL 2010 Conference*, pages 297–305.

[Brennan and Clark, 1996] Brennan, S. E. and Clark, H. H. (1996). Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22(6):1482.

[Browne et al., 2012] Browne, C. B., Powley, E., Whitehouse, D., Lucas, S. M., Cowling, P. I., Rohlfshagen, P., Tavener, S., Perez, D., Samothrakis, S., and Colton, S. (2012). A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43.

[Bunt, 2006] Bunt, H. (2006). Dimensions in dialogue act annotation. In *Proceedings LREC 2006: Fifth International Conference on Language Resources and Evaluation*, pages 1444–1448. ELRA.

[Bunt et al., 2012a] Bunt, H., Alexandersson, J., Choe, J.-W., Fang, A. C., Hasida, K., Petukhova, V., Popescu-Belis, A., and Traum, D. (2012a). Iso 24617-2: A semantically-based standard for dialogue annotation. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 430–437.

[Bunt et al., 2012b] Bunt, H., Alexandersson, J., Choe, J.-W., Fang, A. C., Hasida, K., Petukhova, V., Popescu-Belis, A., and Traum, D. R. (2012b). Iso 24617-2: A semantically-based standard for dialogue annotation. In *LREC*, pages 430–437.

[Bunt and Prasad, 2016] Bunt, H. and Prasad, R. (2016). Iso dr-core (iso 24617-8): Core concepts for the annotation of discourse relations. In *Proceedings 12th joint ACL-ISO workshop on interoperable semantic annotation (ISA-12)*, pages 45–54.

[Campbell and Scherer, 2010] Campbell, N. and Scherer, S. (2010). Comparing measures of synchrony and alignment in dialogue speech timing with respect to turn-taking activity. In *Eleventh Annual Conference of the International Speech Communication Association*.

[Cherny et al., 1999] Cherny, L. et al. (1999). Conversation and community: Chat in a virtual world. Technical report.

[Cho et al., 2014] Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

[Colby et al., 1972] Colby, K. M., Hilf, F. D., Weber, S., and Kraemer, H. C. (1972). Turing-like indistinguishability tests for the validation of a computer simulation of paranoid processes. *Artificial Intelligence*, 3:199–221.

[Davies, 2007] Davies, B. L. (2007). Grice's cooperative principle: meaning and rationality. *Journal of pragmatics*, 39(12):2308–2331.

[Devlin et al., 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[Dozat and Manning, 2016] Dozat, T. and Manning, C. D. (2016). Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

[Dušek et al., 2018] Dušek, O., Novikova, J., and Rieser, V. (2018). Findings of the e2e nlg challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 322–328.

[Edelsky, 1981] Edelsky, C. (1981). Who's got the floor? *Language in society*, 10(3):383–421.

[Eric and Manning, 2017] Eric, M. and Manning, C. D. (2017). A copy-augmented sequence-to-sequence architecture gives good performance on task-oriented dialogue. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 468–473.

[Fandrianto and Eskenazi, 2012] Fandrianto, A. and Eskenazi, M. (2012). Prosodic entrainment in an information-driven dialog system. In *Thirteenth Annual Conference of the International Speech Communication Association*.

[Fillmore, 1968] Fillmore, C. J. (1968). The case for case. *Universals in Linguistic Theory*.

[Finch et al., 2020] Finch, S. E., Finch, J. D., Ahmadvand, A., Dong, X., Qi, R., Sahijwani, H., Volokhin, S., Wang, Z., Wang, Z., Choi, J. D., et al. (2020). Emora: An inquisitive social chatbot who cares for you. *arXiv preprint arXiv:2009.04617*.

[Gandhe and Traum, 2010] Gandhe, S. and Traum, D. (2010). I ' ve said it before, and i ' ll say it again: An empirical investigation of the upper bound of the selection approach to dialogue. In *Proceedings of the SIGDIAL 2010 Conference*, pages 245–248.

[Gervits et al., 2019] Gervits, F., Leuski, A., Bonial, C., Gordon, C., and Traum, D. (2019). A Classification-Based Approach to Automating Human-Robot Dialogue. In *Proceedings of International Workshop on Spoken Dialog System Technology (IWSDS)*, page 12, Siracusa, Italy.

[Giles, 1979] Giles, H. (1979). Accommodation theory: Optimal levels of convergence. *Language and social psychology*, pages 45–65.

[Giles, 2016] Giles, H. (2016). Communication accommodation theory. *The international encyclopedia of communication theory and philosophy*, pages 1–7.

[Giles et al., 1991] Giles, H., Coupland, N., and Coupland, I. (1991). Accommodation theory: Communication, context, and. *Contexts of accommodation: Developments in applied sociolinguistics*, 1.

[Giles et al., 1987] Giles, H., Mulac, A., Bradac, J. J., and Johnson, P. (1987). Speech accommodation theory: The first decade and beyond. *Annals of the International Communication Association*, 10(1):13–48.

[Grice, 1975] Grice, H. P. (1975). Logic and conversation. In *Speech acts*, pages 41–58. Brill.

[Grosz and Sidner, 1986] Grosz, B. J. and Sidner, C. L. (1986). Attention, intentions, and the structure of discourse. *Computational linguistics*, 12(3):175–204.

[Gu et al., 2016] Gu, J., Lu, Z., Li, H., and Li, V. O. (2016). Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640.

[Gunasekara et al., 2020] Gunasekara, C., Kim, S., D'Haro, L. F., Rastogi, A., Chen, Y.-N., Eric, M., Hedayatnia, B., Gopalakrishnan, K., Liu, Y., Huang, C.-W., et al. (2020). Overview of the ninth dialog system technology challenge: Dstc9. *arXiv preprint arXiv:2011.06486*.

[Herring et al., 2013] Herring, S. C., Stein, D., and Virtanen, T. (2013). *Pragmatics of computer-mediated communication*, volume 94. De Gruyter Mouton Berlin.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

[Hu et al., 2017] Hu, Z., Yang, Z., Liang, X., Salakhutdinov, R., and Xing, E. P. (2017). Toward controlled generation of text. In *International Conference on Machine Learning*, pages 1587–1596.

[Huang et al., 2018a] Huang, C., Zaïane, O., Trabelsi, A., and Dziri, N. (2018a). Automatic dialogue generation with expressed emotions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 49–54, New Orleans, Louisiana. Association for Computational Linguistics.

[Huang et al., 2018b] Huang, C., Zaiane, O. R., Trabelsi, A., and Dziri, N. (2018b). Automatic dialogue generation with expressed emotions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 49–54.

[Ippolito et al., 2018] Ippolito, D., Kriz, R., Sedoc, J., Kustikova, M., Callison-Burch, C., Kriz, R., Miltsakaki, E., Apidianaki, M., Callison-Burch, C., Hewitt, J., et al. (2018). Comparison of diverse decoding methods from conditional language models. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. Association for Computational Linguistics.

[Jokinen and McTear, 2009] Jokinen, K. and McTear, M. (2009). Spoken dialogue systems. *Synthesis Lectures on Human Language Technologies*, 2(1):1–151.

[Jurafsky, 1997] Jurafsky, D. (1997). Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *www. dcs. shef. ac. uk/nlp/amities/files/bib/ics-tr-97-02. pdf*.

[Jurafsky and Martin, 2008] Jurafsky, D. and Martin, J. H. (2008). Speech and language processing: An introduction to speech recognition, computational linguistics and natural language processing. *Upper Saddle River, NJ: Prentice Hall*.

[Kawano et al., 2019] Kawano, S., Yoshino, K., and Nakamura, S. (2019). Neural conversation model controllable by given dialogue act based on adversarial learning and label-aware objective. In *Proceedings of the 12th International Conference on Natural Language Generation*.

[Keizer and Rieser, 2017] Keizer, S. and Rieser, V. (2017). Towards learning transferable conversational skills using multi-dimensional dialogue modelling. In *Proc. of SEMDIAL*, pages 158–159.

[Kim et al., 2010] Kim, K., Lee, C., Lee, D., Choi, J., Jung, S., and Lee, G. G. (2010). Modeling confirmations for example-based dialog management. In *2010 IEEE Spoken Language Technology Workshop*, pages 324–329.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Kong et al., 2019] Kong, X., Li, B., Neubig, G., Hovy, E., and Yang, Y. (2019). An adversarial approach to high-quality, sentiment-controlled neural dialogue generation. In *AAAI 2019 Workshop on Reasoning and Learning for Human-Machine Dialogues (DEEP-DIAL 2019)*, Honolulu, Hawaii.

[Kusner et al., 2015] Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document distances. In *International conference on machine learning*, pages 957–966.

[Lafferty et al., 2001] Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

[Lee et al., 2009] Lee, C., Lee, S., Jung, S., Kim, K., Lee, D., and Lee, G. G. (2009). Correlation-based query relaxation for example-based dialog modeling. In *2009 IEEE Workshop on Automatic Speech Recognition Understanding*, pages 474–478.

[Levitan, 2013] Levitan, R. (2013). Entrainment in spoken dialogue systems: Adopting, predicting and influencing user behavior. In *Proceedings of the 2013 NAACL HLT Student Research Workshop*, pages 84–90.

[Levitan, 2020] Levitan, R. (2020). Developing an integrated model of speech entrainment. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*.

[Levitan et al., 2015] Levitan, R., Benus, S., Gravano, A., and Hirschberg, J. (2015). Entrainment and turn-taking in human-human dialogue. In *2015 AAAI spring symposium series*.

[Li et al., 2016a] Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, W. B. (2016a). A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.

[Li et al., 2016b] Li, J., Galley, M., Brockett, C., Spithourakis, G. P., Gao, J., and Dolan, W. B. (2016b). A persona-based neural conversation model. *CoRR*, abs/1603.06155.

[Li et al., 2017a] Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A., and Jurafsky, D. (2017a). Adversarial learning for neural dialogue generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2157–2169.

[Li et al., 2017b] Li, Y., Su, H., Shen, X., Li, W., Cao, Z., and Niu, S. (2017b). Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

[Lin and Och, 2004] Lin, C.-Y. and Och, F. J. (2004). Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612.

[Liu et al., 2016] Liu, C.-W., Lowe, R., Serban, I., Noseworthy, M., Charlin, L., and Pineau, J. (2016). How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proc. of EMNLP*, pages 2122–2132.

[Lukin et al., 2018] Lukin, S., Gervits, F., Hayes, C., Moolchandani, P., Leuski, A., Rogers III, J. G., Amaro, C. S., Marge, M., Voss, C., and Traum, D. (2018). Scoutbot: A dialogue system for collaborative navigation. In *Proceedings of ACL 2018, System Demonstrations*.

[Luong et al., 2015] Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

[Mann and Thompson, 1988] Mann, W. C. and Thompson, S. A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

[Marge et al., 2016] Marge, M., Bonial, C., Pollard, K. A., Artstein, R., Byrne, B., Hill, S. G., Voss, C., and Traum, D. (2016). Assessing agreement in human-robot dialogue strategies: A tale of two wizards. In *International Conference on Intelligent Virtual Agents*, pages 484–488. Springer.

[Meguro et al., 2010] Meguro, T., Higashinaka, R., Minami, Y., and Dohsaka, K. (2010). Controlling listening-oriented dialogue using partially observable markov decision processes. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 761–769.

[Mikolov et al., 2011] Mikolov, T., Kombrink, S., Burget, L., Černockỳ, J., and Khudanpur, S. (2011). Extensions of recurrent neural network language model. In *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5528–5531. IEEE.

[Mizukami et al., 2016] Mizukami, M., Yoshino, K., Neubig, G., Traum, D., and Nakamura, S. (2016). Analyzing the effect of entrainment on dialogue acts. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*.

[Mosier, 1951] Mosier, C. I. (1951). I. problems and designs of cross-validation 1. *Educational and Psychological Measurement*, 11(1):5–11.

[Nasir et al., 2019] Nasir, M., Chakravarthula, S. N., Baucom, B. R., Atkins, D. C., Georgiou, P., and Narayanan, S. (2019). Modeling Interpersonal Linguistic Coordination in Conversations Using Word Mover ' s Distance. In *Proc. Interspeech 2019*, pages 1423–1427.

[Natale, 1975] Natale, M. (1975). Convergence of mean vocal intensity in dyadic communication as a function of social desirability. *Journal of Personality and Social Psychology*, 32(5):790.

[Nenkova et al., 2008] Nenkova, A., Gravano, A., and Hirschberg, J. (2008). High frequency word entrainment in spoken dialogue. In *Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies: Short papers*, pages 169–172. Association for Computational Linguistics.

[Niederhoffer and Pennebaker, 2002] Niederhoffer, K. G. and Pennebaker, J. W. (2002). Linguistic style matching in social interaction. *Journal of Language and Social Psychology*, 21(4):337–360.

[Nivre, 2010] Nivre, J. (2010). Dependency parsing. *Language and Linguistics Compass*, 4(3):138–152.

[Odena, 2016] Odena, A. (2016). Semi-supervised learning with generative adversarial networks. *arXiv preprint arXiv:1606.01583*.

[Oh and Rudnicky, 2000] Oh, A. H. and Rudnicky, A. I. (2000). Stochastic language generation for spoken dialogue systems. In *ANLP-NAACL 2000 Workshop: Conversational Systems*.

[Oh and Rudnicky, 2002] Oh, A. H. and Rudnicky, A. I. (2002). Stochastic natural language generation for spoken dialog systems. *Computer Speech & Language*, 16(3-4):387–407.

[Olsson et al., 1982] Olsson, U., Drasgow, F., and Dorans, N. J. (1982). The polyserial correlation coefficient. *Psychometrika*, 47(3):337–347.

[Papineni et al., 2002] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

[Paranjape et al., 2020] Paranjape, A., See, A., Kenealy, K., Li, H., Hardy, A., Qi, P., Sadagopan, K. R., Phu, N. M., Soylu, D., and Manning, C. D. (2020). Neural generation meets real people: Towards emotionally engaging mixed-initiative conversations. *arXiv preprint arXiv:2008.12348*.

[Pele and M.Werman, 2009] Pele, O. and M.Werman (2009). Fast and robust earth mover's distances. In *Proceedings of the 12th International Conference on Computer Vision*, pages 460–467. IEEE.

[Pele and Werman, 2008] Pele, O. and Werman, M. (2008). A linear time histogram metric for improved sift matching. In *European conference on computer vision*, pages 495–508. Springer.

[Peng et al., 2019] Peng, Y., Fang, Y., Xie, Z., and Zhou, G. (2019). Topic-enhanced emotional conversation generation with attention mechanism. *Knowledge-Based Systems*, 163:429–437.

[Pichl et al., 2018] Pichl, J., Marek, P., Konrád, J., Matulík, M., Nguyen, H. L., and Šedivỳ, J. (2018). Alquist: The alexa prize socialbot. *arXiv preprint arXiv:1804.06705*.

[Pickering and Garrod, 2006] Pickering, M. J. and Garrod, S. (2006). Alignment as the basis for successful communication. *Research on Language and Computation*, 4(2-3):203–228.

[Prasad and Bunt, 2015] Prasad, R. and Bunt, H. (2015). Semantic relations in discourse: The current state of iso 24617-8. In *Proceedings of the 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-11)*.

[Ranzato et al., 2015] Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. (2015). Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

[Ranzato et al., 2016] Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. (2016). Sequence level training with recurrent neural networks. In Bengio, Y. and LeCun, Y., editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

[Reinsch Jr et al., 2008] Reinsch Jr, N. L., Turner, J. W., and Tinsley, C. H. (2008). Multicommunicating: A practice whose time has come? *Academy of Management Review*, 33(2):391–403.

[Reitter and Moore, 2007] Reitter, D. and Moore, J. D. (2007). Predicting success in dialogue.

[Ritter et al., 2011] Ritter, A., Cherry, C., and Dolan, W. B. (2011). Data-driven response generation in social media. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 583–593, Edinburgh, Scotland, UK. Association for Computational Linguistics.

[Roller et al., 2020] Roller, S., Boureau, Y.-L., Weston, J., Bordes, A., Dinan, E., Fan, A., Gunning, D., Ju, D., Li, M., Poff, S., et al. (2020). Open-domain conversational agents: Current progress, open problems, and future directions. *arXiv preprint arXiv:2006.12442*.

[Rubner et al., 1998] Rubner, Y., Tomasi, C., and Guibas, L. J. (1998). A metric for distributions with applications to image databases. In *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)*, pages 59–66. IEEE.

[Sacks, 1992] Sacks, H. (1992). Lectures on conversation: Volume i. *Malden, Massachusetts: Blackwell*.

[Salimans et al., 2016] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. (2016). Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242.

[Sankar et al., 2019] Sankar, C., Subramanian, S., Pal, C., Chandar, S., and Bengio, Y. (2019). Do neural dialog systems use the conversation history effectively? an empirical study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 32–37.

[Searle, 1965] Searle, J. R. (1965). What is a speech act. *Perspectives in the philosophy of language: a concise anthology*, 2000:253–268.

[Sennrich et al., 2016] Sennrich, R., Haddow, B., and Birch, A. (2016). Improving neural machine translation models with monolingual data. In *Proc. ACL*, volume 1, pages 86–96.

[Serban et al., 2016] Serban, I. V., Sordoni, A., Bengio, Y., Courville, A. C., and Pineau, J. (2016). Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, pages 3776–3784.

[Shen and Feng, 2020] Shen, L. and Feng, Y. (2020). CDL: Curriculum dual learning for emotion-controllable response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 556–566, Online. Association for Computational Linguistics.

[Shi and Huang, 2019] Shi, Z. and Huang, M. (2019). A deep sequential model for discourse parsing on multi-party dialogues. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7007–7014.

[Shibata et al., 2016] Shibata, T., Egashira, Y., and Kurohashi, S. (2016). Chat-like conversational system based on selection of reply generating module with reinforcement learning. In *Situated Dialog in Speech-Based Human-Computer Interaction*, pages 63–69. Springer.

[Song et al., 2018] Song, Y., Te Li, C., Nie, J. Y., Zhang, M., Zhao, D., and Yan, R. (2018). An ensemble of retrieval-based and generation-based human-computer conversation systems. In *27th International Joint Conference on Artificial Intelligence, IJCAI 2018*, pages 4382–4388. International Joint Conferences on Artificial Intelligence.

[Stolcke et al., 2000] Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Van Ess-Dykema, C., and Meteer, M. (2000). Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339–373.

[Sun et al., 2018] Sun, X., Chen, X., Pei, Z., and Ren, F. (2018). Emotional human machine conversation generation based on seqgan. In *Proc. of ACII Asia*, pages 1–6. IEEE.

[Sutskever et al., 2014] Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

[Taboada and Mann, 2006] Taboada, M. and Mann, W. C. (2006). Applications of rhetorical structure theory. *Discourse studies*, 8(4):567–588.

[Tao et al., 2017] Tao, C., Mou, L., Zhao, D., and Yan, R. (2017). Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems. *arXiv preprint arXiv:1701.03079*.

[Tian et al., 2017] Tian, Z., Yan, R., Mou, L., Song, Y., Feng, Y., and Zhao, D. (2017). How to make context more useful? an empirical study on context-aware neural conversational models. In *Proc. of ACL*, volume 2, pages 231–236.

[Traum et al., 2018] Traum, D., Henry, C., Lukin, S., Artstein, R., Gervits, F., Pollard, K., Bonial, C., Lei, S., Voss, C., Marge, M., et al. (2018). Dialogue structure annotation for multi-floor interaction. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

[Traum and Nakatani, 1999] Traum, D. and Nakatani, C. H. (1999). A two-level approach to coding dialogue for discourse structure: activities of the 1998 dri working group on higher-level structures. In *Towards Standards and Tools for Discourse Tagging*.

[Traum, 1999] Traum, D. R. (1999). Speech acts for dialogue agents. In *Foundations of rational agency*, pages 169–201. Springer.

[Tuan and Lee, 2019] Tuan, Y.-L. and Lee, H.-Y. (2019). Improving conditional sequence generative adversarial networks by stepwise evaluation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

[Ultes et al., 2017] Ultes, S., Barahona, L. M. R., Su, P.-H., Vandyke, D., Kim, D., Casanueva, I., Budzianowski, P., Mrkšić, N., Wen, T.-H., Gasic, M., et al. (2017). Pydial: A multi-domain statistical dialogue system toolkit. In *Proceedings of ACL 2017, System Demonstrations*, pages 73–78.

[Uzêda et al., 2010] Uzêda, V. R., Pardo, T. A. S., and Nunes, M. D. G. V. (2010). A comprehensive comparative evaluation of rst-based summarization methods. *ACM Transactions on Speech and Language Processing (TSLP)*, 6(4):1–20.

[Verberne et al., 2007] Verberne, S., Boves, L., Oostdijk, N., and Coppen, P. (2007). Discourse-based answering of why-questions.

[Verschueren, 2005] Verschueren, J. (2005). Pragmatics. In *The Routledge Companion to Semiotics and Linguistics*, pages 99–110. Routledge.

[Vinyals and Le, 2015] Vinyals, O. and Le, Q. (2015). A neural conversational model. *arXiv preprint arXiv:1506.05869*.

[Ward and Litman, 2007] Ward, A. and Litman, D. (2007). Measuring convergence and priming in tutorial dialog. *University of Pittsburgh*.

[Webber et al., 2012] Webber, B., Egg, M., and Kordoni, V. (2012). Discourse structure and language technology. *Natural Language Engineering*, 18(4):437.

[Weiss, 2020] Weiss, B. (2020). Talker quality in interactive scenarios. In *Talker Quality in Human and Machine Interaction*, pages 67–106. Springer.

[Weizenbaum, 1966] Weizenbaum, J. (1966). Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45.

[Wen et al., 2015a] Wen, T.-H., Gasic, M., Kim, D., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S. (2015a). Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking. In *Proc. of SIGDIAL*, pages 275–284.

[Wen et al., 2015b] Wen, T.-H., Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S. (2015b). Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

[Williams et al., 2008] Williams, J. D., Poupart, P., and Young, S. (2008). Partially observable markov decision processes with continuous observations for dialogue management. In *Recent Trends in Discourse and Dialogue*, pages 191–217. Springer.

[Williams, 1992] Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

[Xing et al., 2017] Xing, C., Wu, W., Wu, Y., Liu, J., Huang, Y., Zhou, M., and Ma, W.-Y. (2017). Topic aware neural response generation. In *Thirty-First AAAI Conference on Artificial Intelligence*.

[Xing et al., 2018] Xing, C., Wu, Y., Wu, W., Huang, Y., and Zhou, M. (2018). Hierarchical recurrent attention network for response generation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[Xu and Rudnicky, 2000] Xu, W. and Rudnicky, A. I. (2000). Task-based dialog management using an agenda. In *ANLP-NAACL 2000 Workshop: Conversational Systems*.

[Yang et al., 2019] Yang, L., Hu, J., Qiu, M., Qu, C., Gao, J., Croft, W. B., Liu, X., Shen, Y., and Liu, J. (2019). A hybrid retrieval-generation neural conversation model. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1341–1350.

[Yoshino and Kawahara, 2015] Yoshino, K. and Kawahara, T. (2015). Conversational system for information navigation based on POMDP with user focus tracking. *Computer Speech & Language*, 34(1):275 – 291.

[Young et al., 2010] Young, S., Gašić, M., Keizer, S., Mairesse, F., Schatzmann, J., Thomson, B., and Yu, K. (2010). The hidden information state model: A practical framework for pomdp-based spoken dialogue management. *Computer Speech & Language*, 24(2):150–174.

[Yu et al., 2017] Yu, L., Zhang, W., Wang, J., and Yu, Y. (2017). Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, pages 2852–2858.

[Zaremba and Sutskever, 2015] Zaremba, W. and Sutskever, I. (2015). Reinforcement learning neural turing machines-revised. *arXiv preprint arXiv:1505.00521*.

[Zhang et al., 2018] Zhang, Y., Galley, M., Gao, J., Gan, Z., Li, X., Brockett, C., and Dolan, B. (2018). Generating informative and diverse conversational responses via adversarial information maximization. In *Advances in Neural Information Processing Systems*, pages 1810–1820.

[Zhang et al., 2020] Zhang, Y., Sun, S., Galley, M., Chen, Y.-C., Brockett, C., Gao, X., Gao, J., Liu, J., and Dolan, B. (2020). DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.

[Zhao et al., 2017] Zhao, T., Zhao, R., and Eskenazi, M. (2017). Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proc. of ACL*, volume 1, pages 654–664.

[Zhou et al., 2019] Zhou, G., Fang, Y., Peng, Y., and Lu, J. (2019). Neural conversation generation with auxiliary emotional supervised models. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(2):1–17.

[Zhou et al., 2018] Zhou, H., Huang, M., Zhang, T., Zhu, X., and Liu, B. (2018). Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[Zhou and Wang, 2018] Zhou, X. and Wang, W. Y. (2018). Mojitalk: Generating emotional responses at scale. In *Proceedings of the 56th Annual Meeting of the*

*Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1128–1137.

# Publication list

## Journal paper (refereed)

1. Seiya Kawano, Koichiro Yoshino and Satoshi Nakamura, Controlled Neural Response Generation by Given Dialogue Acts Based on Label-aware Adversarial Learning, Transaction of The Japanese Society for Artificial Intelligence, Vol.36, No.4, 2021.

2. Seiya Kawano, Masahiro Mizukami, Koichiro Yoshino, Satoshi Nakamura, Entrainable Neural Conversation Model Based on Reinforcement Learning, IEEE Access, IEEE, Vol.8, pp.178283-178294, 2020.

## International conference paper (refereed)

1. Seiya Kawano, Koichiro Yoshino, Traum David, Satoshi Nakamura, Dialogue Structure Parsing on Multi-floor Dialogue Based on Multi-Task Learning, The 1st RobotDial Workshop on Dialogue Models for Human-Robot Interaction (ROBOTDIAL) , IJCAI-PRICAI, 2021.

2. Seiya Kawano, Koichiro Yoshino, Satoshi Nakamura, Neural Conversation Model Controllable by Given Dialogue Acts Based on Adversarial Learning and Label-aware Objective, The 12th International Conference on Natural Language Generation (INLG 2019), pp.198-207, 2019.

3. Seiya Kawano, Koichiro Yoshino, Yu Suzuki, Satoshi Nakamura, Dialogue Act Classification in Reference Interview Using Convolutional Neural Network with Byte Pair Encoding, International Workshop on Spoken Dialogue System Technology (IWSDS 2018), 2018. (Best paper award runner-up)

## Domestic conference paper

1. 河野誠也, 吉野幸一郎, David Traum, 中村哲, マルチタスク学習に基づいた複数フロアの対話構造の自動解析, 言語・音声理解と対話処理研究会 (第 91 回), 人工知能学会, 2021. (研究会優秀賞)

2. 中野佑哉, 河野誠也, 吉野幸一郎, 須藤克仁, 中村哲, 問い返し質問文生成によって曖昧性解消を行う質問応答システム, 言語処理学会年次大会, 2021.

3. 邊士名朝日, 野中尋史, 河野誠也, 谷川英和, 大域的・局所的エントロピーに基づいた特許文書中からの効果述語項構造の自動抽出, 言語処理学会年次大会, 2021.

4. 中野佑哉, 河野誠也, 吉野幸一郎, 須藤克仁, 中村哲, 文構造に基づく質問文への曖昧性付与と質問生成, 言語・音声理解と対話処理研究会 (SIG-SLUD), 2020.

5. 中野佑哉, 河野誠也, 吉野幸一郎, 中村哲, 対話によって曖昧性解消を行う質問応答, 情報処理学会第 244 回自然言語処理研究会, 2020-NL-244, pp.1-5, 2020.

6. 河野誠也, 吉野幸一郎, 中村哲, 文脈に対して同調的応答を生成するニューラル会話モデルの検討, 人工知能学会第 34 回全国大会 (JSAI2020), 3Rin4, pp. 1-4, 2020.

7. 河野誠也, 吉野幸一郎, 中村哲, 対話行為を用いた制御可能なニューラル対話モデルの検討, 人工知能学会第 33 回全国大会 (JSAI2019), 3Rin2-27, pp. 1-4, 2019.

8. 河野誠也, 吉野幸一郎, 中村哲, 条件付き敵対的生成ネットワークを用いたデータ拡張による対話行為分類法の検討, 情報処理学会第 125 回音声言語情報処理研究会 (SIG-SLP), 2018-SLP-125, pp. 1-6, 2018. (Yahoo!-japan 賞：学生優秀論文賞)

9. 河野誠也, 吉野幸一郎, 鈴木優, 中村哲. 対話行為予測に基づいたレファレンスインタビューのモデル化とその分析, 情報処理学会第 234 回自然言語処理研究会 (SIG-NL), pp.1-6, 2017.

10. 河野誠也, 吉野幸一郎, 鈴木優, 中村哲. CNN を用いたレファレンス・インタビューにおける対話行為の推定, 情報処理学会第 117 回音声言語情報処理研究会 (SIG-SLP), pp.1-6, 2017.

## Book chapter

1. Seiya Kawano, Koichiro Yoshino, Yu Suzuki, Satoshi Nakamura, Dialogue Act Classification in Reference Interview Using Convolutional Neural Network with Byte Pair Encoding, Lecture Notes in Electrical Engineering (LNEE) 579, Springer, pp.17-25, 2019.